**Final Project**

**IST687- Team 2**

Brandon Liunoras

Diana Bradberry

Jonathan David

Timothy Wasserman

12/20/2022

Table of Contents

[**1. Introduction**](#_cns8dltl69xw) **3**

[1.1 Project Background and Description](#_gjf2yqo836ft) 3

[1.2 Project Scope and Context of Analysis](#_tqmjz5ksetcr) 4

[**2. Business Questions**](#_zcniiq88tn0l) **4**

[**3. Data Management Methods**](#_xs648i4d42x5) **5**

[3.1 Data Acquisition](#_j1mh7lfyxv1y) 5

[3.2 Data Dictionary/Initial Quality Assessment](#_dahrk6pg9666) 6

[3.3 Data Cleansing](#_jsefs420n637) 7

[3.4 Regression Methodology](#_gvnuqujswsrt) 8

[**4. Results**](#_46cdhc4gc939) **10**

[4.1 General Statistics](#_e2friyofe86a) 10

[4.2 Individual Regression Model Results](#_rf1my6dgbn7b) 12

[4.3 Multiple Regression Model Results](#_4dkq5mqei3tk) 13

[**5. Conclusions**](#_sb1aueh8trn4) **17**

[5.1 Final Conclusion/Remarks](#_8aombqirtk2s) 17

[5.2 Limitations](#_5z0pqzmypxt1) 18

[**6. Appendix - Code/Plots/Diagrams**](#_lkvlzfm81zav) **19**

[**7. References**](#_d05rkqdfjhp5) **41**

# 1. Introduction

## 1.1 Project Background and Description

This project is an exercise in taking a dataset and using it to gauge what factors are able to provide insight into factors likely influencing the California rental market and what we can learn from the data.

In fall 2022, Team 2 in an introductory graduate-level data science course at Syracuse University (IST 687) were tasked with acquiring and analyzing a publicly available (or privately available, by consent) dataset. After considering several candidate datasets, the Team decided to proceed with a dataset of San Francisco Bay Area rental housing listings on Craigslist (Pennington, 2018). Each record in the dataset contained the year, the price of the rental, information on location (neighborhood, city, county), and attributes of the rental unit including beds, baths, and square feet. Information on year, price, location, and beds was complete or nearly complete across records. Baths, and square feet, had many missing values (i.e., NA) which could be remedied by imputation.

Historically, rental prices have been particularly high in the San Francisco Bay and its surrounding areas, in both an absolute sense and relative to the size and features of the apartment (Badger, 2016; Boeing et al., 2020). These trends have been persistent; a newspaper article from December 2022 examined the lack of affordable housing nationally and in select cities including San Francisco (Williams, 2022). Elevated prices present challenges for those seeking housing in the region, as low-income and even middle-income individuals and families struggle with costs. Competition for desirable apartments leads to further scarcity of viable options (Badger, 2016; Boeing et al., 2020). While home ownership—in contrast to renting—is viewed as an ideal to aspire to in the U.S., the share of households living in rentals has remained close to one-third of all households (Boeing & Waddell, 2017). The percentage of rentals in relation to all living arrangements is forecast to rise through 2050, reflecting several factors including a desire for mobility, insufficient income or qualification to gain a mortgage, and concerns that home ownership will not yield investment gains commensurate with the purchase price (Nelson, 2016). Even if these prognostications overstate the role of renting in the future, rental housing arrangements can be expected to serve a large portion of the U.S. population in the future.

The Craigslist data utilized for this study are particularly suited to an analysis of pricing dynamics, containing near-complete price and geospatial information (Boeing & Waddell, 2017) as well as data on other rental unit attributes. Given the relatively high rental pricing in the San Francisco Bay Area, study of pricing patterns can provide insight into how prices vary by location and apartment attributes.

Along with explaining variation in pricing, research may also provide insight into the typifiers of more-affordable housing options. Although Bay Area prices are high overall, research may find that there are pockets of affordability. Investigation can also uncover if location, or instead rental unit amenities and sizes, factor more heavily in prices. With records from years 2003, 2008, 2013, and 2018—spanning fifteen years—the data also enable an inspection and estimate of price increase over time. These research objectives are stated as the study’s business/research questions

## 

## 1.2 Project Scope and Context of Analysis

The scope of this project encompasses data gathered between the years 2003 and 2018, regarding rental properties in San Francisco and surrounding areas. The data provides insight into the change in rental property price overtime and what factors may be contributing to the changes in price we see shape over time. Using this data, the goal is to analyze and pinpoint main rent drivers and create discussion on what can be done to stabilize or even decrease rent prices.

# 2. Business Questions

Within this report, we research living rentals and their prices within the state of California. Below are the following questions that we had researched to help analyze the California rent dataset:

* How do rent prices change depending on neighborhood and county?
* Does the availability of rentals within a location affect price?
* What are the main rent drivers of rental prices in California?

# 

# 

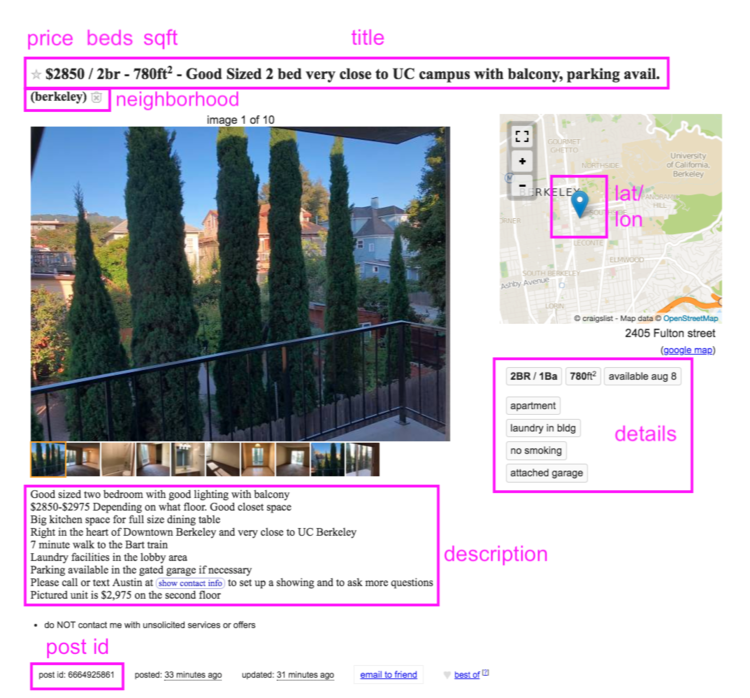
# 

# 3. Data Management Methods

## 3.1 Data Acquisition

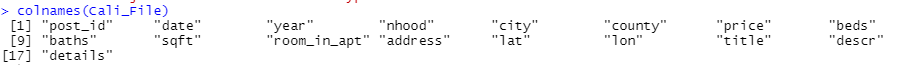
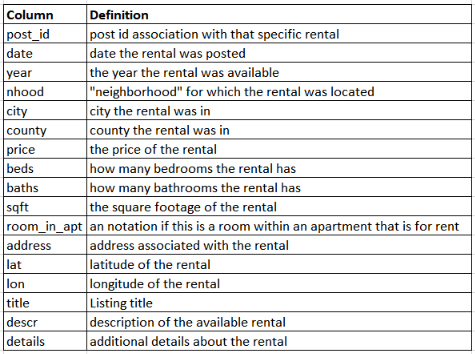
The data that was acquired is from a publicly available website of Craigslist rental housing posts (Pennington, 2018). The data on the website had been cleaned such that the geographical and location variables in the data were consistent across records and within the dataset as a whole (Pennington, 2018). While the original dataset spanned years 2000 through 2018 and had over 200,000 records, the data was filtered to years 2003, 2008, 2013, and 2018. This yielded a smaller dataset of records that spanned nearly the same number of years as the original dataset, yet was much more workable for this study.

Data for each record was sourced from an archived URL of the Craigslist post. The date and year fields came from the archive date of the URL. Most of the other variables are detailed in the following image from Pennington (2018). Though descriptions for the city and county variables are not explicitly provided, a review of Pennington (2018) indicates that it is reasonable to assume that they are aggregations of the variable “nhood” (neighborhood.)

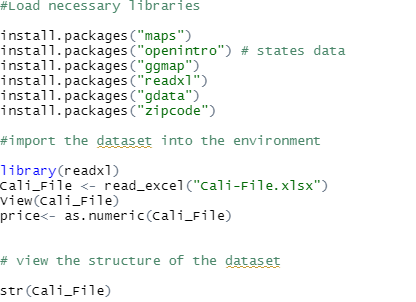


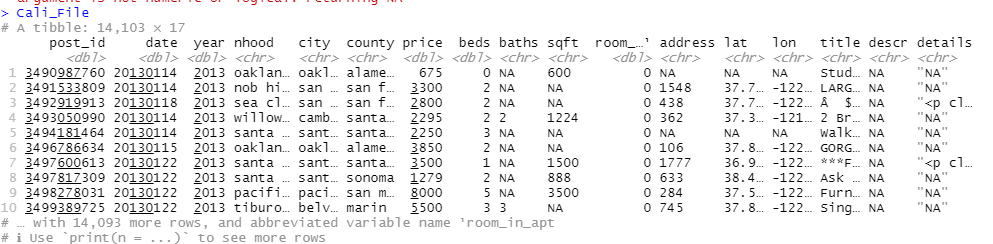
## 3.2 Data Dictionary/Initial Quality Assessment

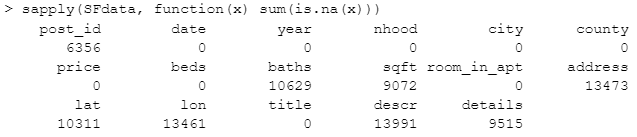
There were seventeen variables in the dataset. These included post\_id, date, year, nhood (neighborhood), city, county, price, beds, baths, sqft, room\_in\_apt, address, lat, lon, title, descr (description), and details. The following diagram contains a table of the dataset’s variables with their respective definitions.



After an initial review of the dataset records, it was found that the observations contained character and double values. It was also found that further preprocessing work was necessary to prepare the data for full analysis. There were many NA values within multiple columns in both the character and the double value columns. Shown below is code the structure of the main file along with NA values in the dataset.



****

****

## 3.3 Data Cleansing

The dataset was cleaned through different methods such as deletion of incomplete records, transformation of observations, and NA interpolation. Records missing the bed variable–an essential measure–were few, and were removed. Cities without records in all four years were also deleted. The final dataset had just over 14,000 records and seventeen variables, including a key outcome measure (price) along with several variables to predict price. The “title” variable contained string data that provided additional representation of rental units.

After initial filtering, inspection showed that only minimal cleansing was needed to transform some of the columns–as they were not fully in the format that we desired them to be. Any transformations were done so that our analysis and data modeling would be simplified for the analysis in later parts of the project.

In order to model rental prices against the other variables in the dataset, imputation was used for NAs on baths and sqft; new columns were created:

SFdata$baths\_na\_interpolation\_i <- na\_interpolation(SFdata$baths)

SFdata$sqft\_na\_interpolation\_i <- na\_interpolation(SFdata$sqft)

To account for the NA values within the variables “lat” and “lon”, the means of these variables were substituted into the NA areas. Since geographies are bounded, a reasonable way to impute is to use the mean of coordinates (by city) based on non-NA cases. Use of an imputation routine risked imputing a lat/lon that was geographically outside the city bounds; hence, non-NA mean lats and longs by city were used for imputation. The code for this substitution is in appendix 6.4.

In addition to cleaning the data, city population was also added to each record to be used as needed. This required use of an external, publicly-available file from the California Department of Finance. Where records did not merge to the Craigslist dataset, some hard coding was required to fill in the NAs. This code is in appendix 6.5.

To prevent issues with linear regression and resulting datasets, where names of variables may derive from county/city/nhood values (e.g. for feature selection like backwards elimination), backslashes and spaces within values were converted to underscores and saved into new variable columns. Code for deriving these alternative designations for county/city/nhood is in appendix 6.6.

## 3.4 Regression Methodology

To provide additional perspective on research question 1, linear modeling of apartment price was undertaken. Three separate models each using a specific type of location category (i.e., neighborhood, county (and city) were fit. Year, number of beds and baths, and square feet were also in each model. R code for each model:

m\_full\_nhood <- lm(formula = price ~ beds + baths\_na\_interpolation\_i + sqft\_na\_interpolation\_i +

nhood\_cln, data=SFdata)

summary(m\_full\_nhood)

m\_full\_city <- lm(formula = price ~ beds + baths\_na\_interpolation\_i + sqft\_na\_interpolation\_i +

city\_cln, data=SFdata)

summary(m\_full\_city)

m\_full\_county <- lm(formula = price ~ beds + baths\_na\_interpolation\_i + sqft\_na\_interpolation\_i +

county\_cln, data=SFdata)

summary(m\_full\_county)

## 

To provide insight for this question, a linear regression model of apartment price was first developed using the full dataset. Linear regression is an appropriate technique for modeling and predicting continuous dependent variables (Saltz & Stanton, 2022). Predictors included year, number of beds and baths, square feet, and neighborhood. Neighborhoods were represented by indicator variables, each capturing the name of a neighborhood:

# Get dummy indicators for nhood\_cln. This will represent each value of nhood\_cln as an indicator # variable. Having explicit indicator variables will allow us to manage each nhood in the # modeling process. https://cran.r-project.org/web/packages/fastDummies/fastDummies.pdf

library(fastDummies)

# table(SFdata$nhood\_cln)

# `santa\_cruz` is the highest-frequency nhood (n=451), and is omitted from dummy construction.

SFdata <- dummy\_cols(SFdata, select\_columns = "nhood\_cln", remove\_most\_frequent\_dummy = TRUE)

str(SFdata)

summary(SFdata)

The dataset (i.e., data frame) was copied into a dataframe of only quantitative variables for regression analysis. An initial model including all predictors and all categories of neighborhood was fit.

# First, run the full model. Then, run feature-selection algorithm.

# first get the full model:

m\_full\_nhood <- lm(price ~ ., data = SFdata\_lm\_vars)

summary(m\_full\_nhood)

The model was subsequently trimmed using a backwards elimination feature selection process.

# Info: https://cran.r-project.org/web/packages/olsrr/vignettes/variable\_selection.html

# https://www.rdocumentation.org/packages/olsrr/versions/0.5.3/topics/ols\_step\_backward\_p

# install.packages("olsrr")

library(olsrr)

library(car)

car::vif

# Apply the backwards elimination routine to ultimately retain significant predictors.

# Eliminate predictors not significant at the 0.05 level.

m\_full\_nhood\_step\_backward <- ols\_step\_backward\_p(m\_full\_nhood, prem = 0.05)

str(m\_full\_nhood\_step\_backward)

m\_full\_nhood\_step\_backward

Results from the backwards elimination process were used to trim non-significant independent variables from the full model. The model was then re-fit using only significant predictors not removed by the backwards elimination procedure. This yielded a final, full-dataset model. The code for running the initial full-dataset model, the backwards elimination procedure, and obtaining the final, full-dataset model from backwards elimination is in appendix 6.7.

To check for the possibility of overfit and guard against the model “memorizing” the data (Saltz & Stanton, 2022, p. 224), a training/validation approach to development of the linear model was also taken. The full dataset was partitioned into 75%/25% random training/validation split. A model was then trained on the 75% training set– i.e., an initial full model, then run through a backwards elimination process to eliminate nonsignificant predictors and arrive at a final model. With the training model created, it was possible to assess the importance of each predictor. It was also possible to run various combinations of predictors through the model to yield an estimated rental price for the given values of predictors.Then, the 25% validation set was run through the final model and predicted values of price were generated. The root mean square (RMS) of the residuals (i.e., actual price minus predicted price) was generated for the validation set, and compared to the RMS for the training model. In this process it is worth noting that the training set may have omitted a few small neighborhoods that were in the test set, and in this way the training model may `under-represent` such neighborhoods. We proceeded, noting this as a possible caveat. The code for running the training/validation model, evaluating the relative importance of predictors, and assessing test set cases with the training model is in appendix 6.8.

To furnish insight into the descriptors and typifiers of rental units within each neighborhood, word clouds were created from the *title* field. The word cloud showed words by size and by centrality in the cloud; larger and more-central words in the cloud occurred most frequently. A word cloud was created for the data as a whole. Additionally, a function was defined that created a world cloud for a specific neighborhood. The minimum count per word could also be specific in the function. Code for the word clouds is in appendix 6.9.

# 4. Results

## 4.1 General Statistics

The complete dataset has 17 variables and 14,103 observations. Measures of central tendency were calculated when looking at the general statistics of the dataset with mean being utilized the most. Important averages to note out of the dataset are the average rent price by county, city, and neighborhood. Full code and statistics is found in the appendix.

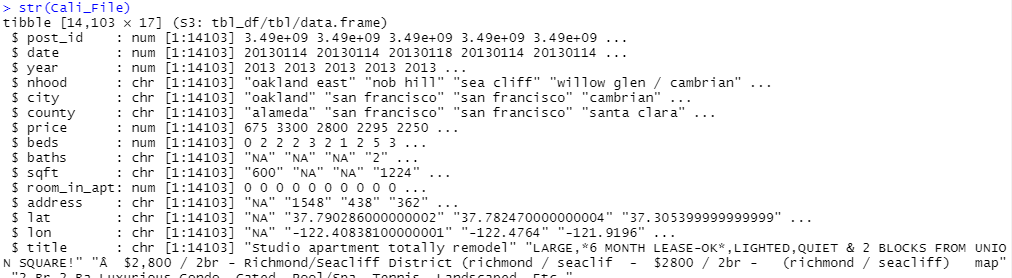
In addition to acquiring the averages across the whole dataset, price comparisons were made across location and timeframes. To start, counties like Napa, Marin, Alameda, and Santa Clara have all seen rental prices double since 2003, while counties like Solano and Contra Costa have seen about a 60% increase in rental prices from 2003 to 2018. For example, in the county of San Francisco, rental prices shifted an average of $1,400 among counties between 2003 and 2018. Since 2003, the rent in Marin and San Francisco counties has trended highest (averaging $2,789 for a standard 3 bed, 2 bath rental in 2003 and $5,027 for the same in 2018), while other counties like San Mateo and Alameda averaged $2,121 for a 3 bed, 2 bath rental in 2003 and $3,663 in 2018 for the same (a $1,542 increase in 15 years).

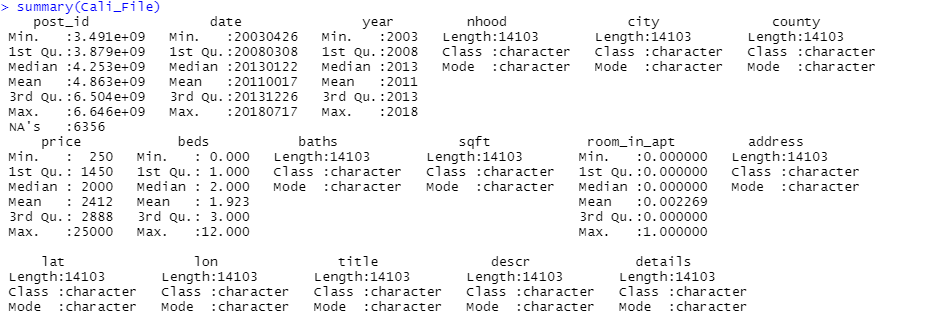
Rent in San Francisco was highest around 2013 where Marin county took the lead, followed by Napa (second), and San Francisco (third). Solano and Santa Cruz counties had the lowest rent in 2003 (Solano County average rent was $1,283 and Santa Cruz county averaged $1,157).

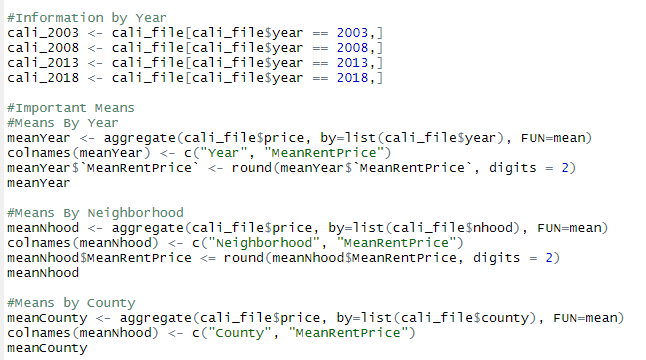
Similar to the county analysis, rent was the highest within neighborhoods nestled within San Francisco county in 2003, with the neighborhood of Portola averaging the highest rent average ($5,900), followed by West Portal/ Forest Hills ($4,458) while Berkeley downtown ($762) had the lowest rent average (possibly because this is mostly student housing for those attending the university), followed by Oakland ($800), located in Alameda County.

The neighborhood specific rental market has seen more significant shifts over the years, between 2003 and 2018, increasing an average of $1,407; or $94 per year. In 2018 the Tiburon/ Belvedere neighborhood within Marin county, jumped into the lead as the highest rental neighborhood (averaging $8,607) pushing West Portal/ Forest Hills into the second highest location ($7,483).

On the other hand, Russian River (in Sonoma) had the lowest rental average in 2018, at about $1,550 for a standard 3 bedroom (however, this is an increase of about $650 from $900 in 2003) and Vallejo/ Benicia (also in Solano) had the second lowest average in 2018, at $1,798 (but, an increase of about $662 from 2003).







Separate regression models were run for neighborhood, county, and city. In each model, at least some of the predictors capturing various locations were statistically significant. These results indicate that apartment prices vary by neighborhood, county, and city. However, the adjusted r-squared varied by model. The respective r-squared values for the neighborhood, county, and city models were .51, .41, and .44. Thus, the neighborhood model captured the most variation in apartment price. This is not surprising, given that neighborhood is a more discrete or granular categorization of locale than city or county.

## 4.2 Individual Regression Model Results

In this section, different variables within the dataset were tested within a linear model against the variable price to see which variables solely affected the rental price. The code that was used reads as a summary of a linear model, with price being the dependent variable, and other columns such as “city” as the independent variable. Certain variables have been omitted such as latitude and longitude as location is more precisely covered with other variables. Full code and results will be displayed within the appendix (add notation). The following is a list of the regression results:

| **Independent Variable** | **Adjusted R-Squared** |
| --- | --- |
| County | 0.1189 |
| City | 0.1538 |
| Neighborhood | 0.2186 |
| Sqft (Square Footage) | 0.02762 |
| Beds | 0.2002 |
| Baths | 0.0177 |

From the regression, all the results of the regression displays that the variables have adjusted R-squared values ranging between 0.02 and 0.23. These regression values are weak and show that one single independent variable does not have a great influence over rental prices. More testing is needed to determine rent drivers.

## 4.3 Multiple Regression Model Results

Summary results from the regression analyses of rental price are shown in the table below. Across models, summary statistics were not sensitive to whether a full, or 75% validation, dataset was used to fit the model. Models were also very similar after removing predictors identified by the backwards elimination procedure. All models, including the final model, accounted for almost 60% of the variation on prices. The RMS error of prediction–a little more than $1,000–was also similar across models. These results demonstrate that even with a model that explains the majority of variation in the outcome, the prediction error can still be substantial.

| **Model\*** | **Adjusted R-squared** | **RMS Residual** |
| --- | --- | --- |
| Full-dataset; all predictors | 0.5895 | $1,030 |
| Full-dataset; predictors remaining after backwards elimination | 0.5886 | $1,031 |
| 75% training model; all predictors | 0.5900 | $1,021 |
| 75% training model; predictors remaining after backwards elimination (Final Model) | 0.5896 | $1,021 |

\* All *p* < .05.

When test set records were run through the final (75% training) model, the RMS error of their predicted values climbed moderately to $1,072. While prediction accuracy for the test set was a little lower than for the training set, the RMS quantities for the models remained within 5% of each other. Given this result, and the fact that overfit with linear regression is typically not a large concern (Saltz & Stanton, 2022), we concluded that the final model was reliable and workable for rental price analysis.

A detailing of the predictors in the final regression model in the table below showed that each is an independent and significant predictor of rental price. The neighborhoods with the top five importance values are included in the table.

| **Predictor\*** | **Estimate** | **Importance (max=100)** |
| --- | --- | --- |
| Year | $112 | 50 |
| Beds | $725 | 73 |
| Baths | $84 | 5 |
| SQFT | $0.30 | 19 |
| Neighborhood: SOMA South Beach | $2,240 | 37 |
| Neighborhood: Pacific Heights | $2,519 | 33 |
| Neighborhood: Russian Hill | $2,421 | 25 |
| Neighborhood: Marina Cow Hollow | $1,939 | 23 |
| Neighborhood: Nob Hill | $1,269 | 18 |

\* All *p* < .05.

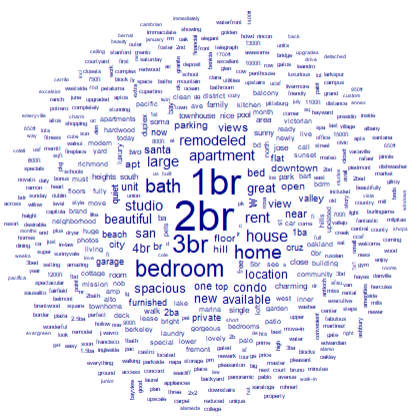
From these data, it is evident that variation in the number of beds in a rental unit, and location, can greatly impact listing price. Locations such as SOMA South Beach, Pacific Heights, and Russian Hill command price premiums roughly equal to three beds, relative to locations that are not predictors in the final model. While many locations were associated with rental premiums that ranged from -$1,000 to $1,500, rent for a few locations including Brentwood Oakley Discovery Bay, Fairfield Vacaville, Pittsburg Antioch, and Vallejo Benicia was over $1,000 less expensive compared to locations not having predictors in the final model.

Year of listing was estimated to have a $112 per year impact on rental price. All else equal, a rental in 2018 that listed for $1,000 in 2003 was estimated to cost (15 x $112 =) $1,680 more. Indeed, year was the second-most-important predictor in the model, demonstrating that the cost of housing has risen substantially over the years in the San Francisco Bay Area. The size of the apartment, in square feet, had a modest impact on list price. With an estimate of 30 cents per square foot, an apartment 100 square feet larger than a very similar unit was estimated to list for $30 more. Relative to other factors, only large swings in square feet have a major impact on price. Finally, the number of baths had little impact on price. Baths also had a low importance score. With most units having one or two baths, the premium for a two-bath apartment was estimated to be only $84 more than a one-bath unit.

Using the final model, apartment prices can be estimated for specific values of year, number of beds and baths, square feet, and neighborhood. Assuming that the model can be extrapolated and applied beyond the 2003 to 2018 training years, current projections of pricing are possible. For example, in 2022, a one-bedroom, one-bath, 1,000 sqft apartment in San Jose East is predicted to list for $1,988. Ten years prior in 2012, the cost estimate for the same apartment was $867. Meanwhile, in 2022, the same apartment in Pacific Heights is estimated to cost over $5,000– an over $3K premium for Pacific Heights compared to San Jose East for an otherwise same unit. Comparing these estimates to current, actual prices for apartments with specific attributes would be a worthwhile exercise to determine if the model still holds.

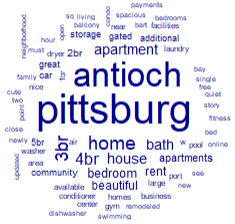
A full listing of the final model, and importance of each predictor, can be found in appendix 6.8.

To graphically summarize the features in the rental units, a word cloud was produced from the titles of the listings:



The most frequently-occurring words were simple attributes of the apartments such as number of bedrooms and the type of unit (e.g., studio; condo, apartment; home). Beyond these words were frequently-occurring adjectives such as “large,” “beautiful,” “remodeled,” and “spacious.” “Parking,” “location,” and “view” were also frequently mentioned. These attributes may help to explain price fluctuation in units that are otherwise similar in beds, baths, size, and location.

The stark difference in listing price by location raised the question of how apartment features might be related to price. For this exercise, separate word clouds were developed for an expensive (Pacific Heights) and inexpensive (Pittsburg Antioch) location. The estimated difference in price for these two locations, net of other apartment attributes, was $3,800.

One conclusion from these word clouds is that units in Pacific Heights tend to be one or two bedrooms, while units in Pittsburg Antioch are three or four bedrooms. The larger number of bedrooms in the latter location, and perhaps their size too, may move their listing price closer to units in Pacific Heights. Further supporting this interpretation was the considerable mention of “home” and “house” for Pittsburg Antioch, implying larger spaces with many bedrooms in this location. In contrast, the types of spaces in the Pacific Heights word cloud tended to be “studio,” “apartment,” and “condo,” apartments that would not be expected to be as large as houses.

Qualitative features of units different somewhat between the two locations. Pacific Heights units included descriptors such as “views,” “location,” “furnished,” and “remodeled.” These features figured less prominently for Pittsburg Antioch, and may account for some of the difference in price by location. “Beautiful” was a somewhat-frequent descriptor for Pittsburg Antioch, though this term is non-specific and may carry less weight in apartment price.

# 5. Conclusions

## 5.1 Final Conclusion/Remarks

We undertook this study to understand the factors influencing rental listing prices in the San Francisco Bay Area. With its notably high housing costs and ongoing price increases, research looking at this region uncovered factors driving prices. Geographic location and housing availability showed a relationship to price. Using a modeling approach, the rental premium associated with specific apartment unit attributes is also estimated.

We took a closer look at whether the availability of rentals in a location influences the rental price- and we found that it does influence the price. If the rental is available in a specific location, the rental price goes up depending on the city and the sqft. If there is no availability, then it does not appear to affect the price of the rental.

A linear model of apartment price was developed. Predictors include year, number of beds and baths, square feet, and neighborhood. Neighborhoods were represented by indicator variables. An initial model including all predictors and all categories of neighborhood was fit. The model was subsequently trimmed using a backwards elimination feature selection process. The provisional final model included year, number of beds and baths, square feet, and many (but not all) neighborhood indicators. Thus, these five predictors are all significant drivers impacting apartment prices. Unsurprisingly, data collected on the location and structural factors of apartments were main drivers of rental prices in San Francisco. These include:

* # of beds
* Location of the rental
* Sqft\* Imputed Data
* Baths\* Imputed Data

Year also impacted price.

These results present useful information for individuals, stakeholders, and other audiences interested in the dynamics underlying Bay Area housing rental prices. For example, the high price premiums commanded in some neighborhoods may render these units unaffordable to those who live there or would like to live there. Those concerned with affordability and accessibility such as governmental agencies and non-governmental organizations (NGOs) should take note of the dramatic variation in pricing by neighborhood. To serve those of modest means living in costly or gentrifying/gentrified neighborhoods, solutions to decrease the rental burden such as housing subsidies, property tax adjustments, and/or expansion of supply may be beneficial. As this study showed, the number of bedrooms in a living unit was positively related to price, and model results showed it was of top importance across all variables that impacted price. This underscores the centrality of bed space in rental unit pricing.

Solutions that serve one stakeholder group may not swerve another. Increased availability of rental units in a given location may moderate or drive down rental prices, which may be counter to the interests of investors, landlords, and property owners whose interest is in fetching maximum price for their rental units. This study may provide useful information to such individuals, helping them to identify factors determining the fair market value of their properties and providing approaches to a research-based accounting of rental unit prices.

## 5.2 Limitations

This study is subject to several limitations. It utilized data from only one geographic location—a large metropolitan area and its surrounding cities in western California. To the extent that rental prices across the U.S. are affected by factors such as location, housing supply and demand, the job market, and quality of schools and services the results of the current study may not generalize to other regions. Similarly, this study did not capture all factors potentially impacting list prices. Accounting for additional rental units attributes and their contexts–from number of windows in the apartment, to age of the unit, to neighborhood safety, to vacancy rates, and quality of public services in the locale–may enable more-thorough and more-accurate estimation of rental factors and prices.

Additionally, the data for this study are sourced only from Craigslist posts. Other sources, such as traditional media (i.e., newspapers and other print materials), social media (e.g., Facebook), and search engines (e.g., Google) are not included in the data. To the extent that patterns of rental unit pricing differ across platforms and from Craigslist, analysis of data from such sources may yield results different from those in this study.

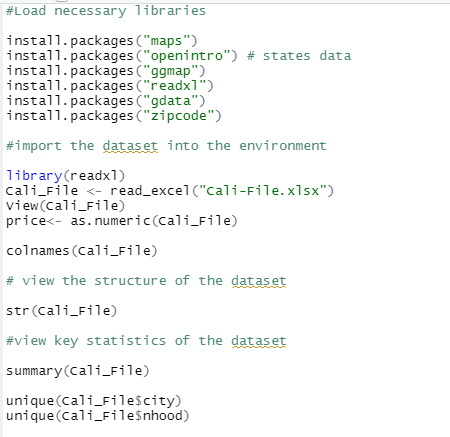
Furthermore, only data up to the year 2018 was used. These findings are not the most representative of rental prices today as it omits an important housing trend: the COVID-19 pandemic. This is an important observation to be made because the data used does not account for the sharp increase in rental prices and dwindling supply of available rentals during this time.

One last limitation is that only the list price of rentals was available. Actual paid rent may have differed from the list price due to such factors as rent control or other subsidies, word-of-mouth advertising or informal lease arrangements, and “bidding wars” for highly desirable units (Badger, 2016; Boeing et al., 2020). Replication of this study in other locations and contexts, using additional sources of data, would shed light on generalizability to the broader U.S. rental housing market.

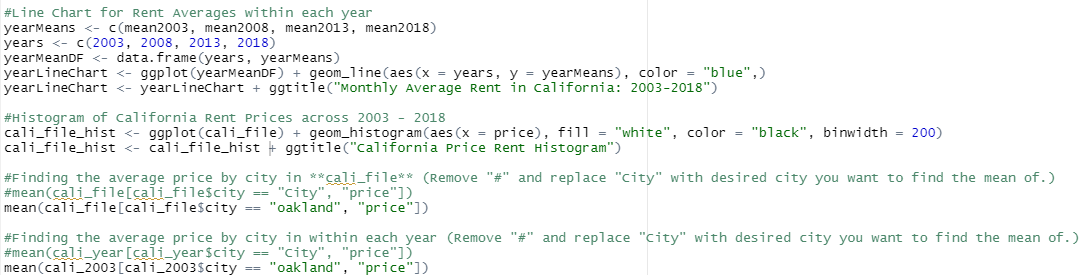
# 6. Appendix - Code/Plots/Diagrams

Code/ additional diagrams used in the analysis:

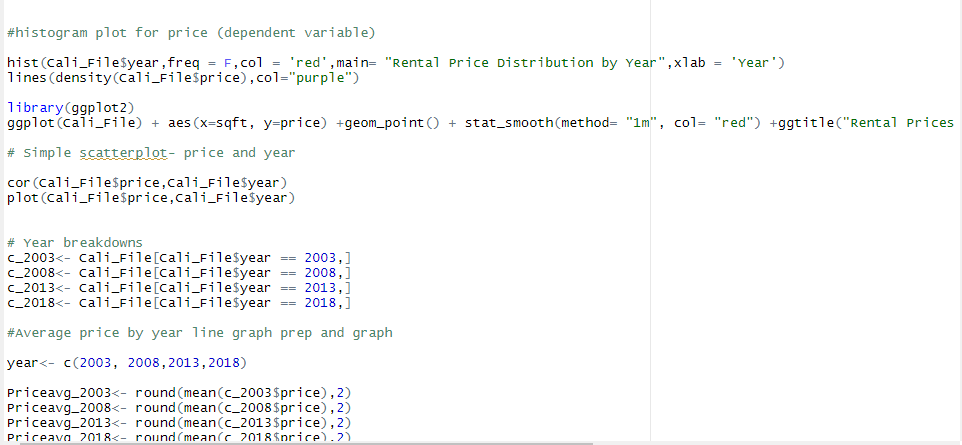
1. Reading in and reviewing the data:



1. Reading in and reviewing the data:



1. Code for descriptive analysis of price:



1. Code for substituting the non-NA mean lat and lon for NA lat and lon, by city:

# Make new dfs for non-NA city lats & lons. Then do means by group and store.

# I chose to keep only needed columns, but the downside is that I used hard-coding to do this so it assumes a particular structure to the DF:

SFdata\_lat\_no\_NAs <- SFdata[!is.na(SFdata$lat), ][,c(5,13)]

SFdata\_lon\_no\_NAs <- SFdata[!is.na(SFdata$lon), ][,c(5,14)]

# Now get means. These are lists and not in dataframe format for merging.

lat\_mean\_by\_city <- tapply(SFdata\_lat\_no\_NAs$lat,SFdata\_lat\_no\_NAs$city,mean)

lon\_mean\_by\_city <- tapply(SFdata\_lon\_no\_NAs$lon,SFdata\_lon\_no\_NAs$city,mean)

# Convert to dataframes. However, city goes into rowname; we need city as a column for merge.

lat\_mean\_by\_city <- as.data.frame(lat\_mean\_by\_city)

lon\_mean\_by\_city <- as.data.frame(lon\_mean\_by\_city)

# Tidyverse`s `rownames\_to\_column` will work.

lat\_mean\_by\_city <- rownames\_to\_column(lat\_mean\_by\_city, var = "city")

lon\_mean\_by\_city <- rownames\_to\_column(lon\_mean\_by\_city, var = "city")

# Now see that the 2 dataframes of means are ready to merge to the SF data, by city.

lat\_mean\_by\_city

lon\_mean\_by\_city

# Merge means to the main dataframe; these means will be substituted for NAs.

SFdata <- merge(SFdata, lat\_mean\_by\_city, by = "city")

SFdata <- merge(SFdata, lon\_mean\_by\_city, by = "city")

# Assign the mean where the native value is NA. In the textbook page 163 we see the ifelse function:

# Name these variables with a `\_city\_mean\_i` suffix to tell us how they were imputed.

SFdata$lat\_city\_mean\_i <- ifelse(is.na(SFdata$lat), SFdata$lat\_mean\_by\_city, SFdata$lat)

SFdata$lon\_city\_mean\_i <- ifelse(is.na(SFdata$lon), SFdata$lon\_mean\_by\_city, SFdata$lon)

1. Code for adding mean city population to each record:

# Here is a URL of CA city population:

# https://dof.ca.gov/wp-content/uploads/Reports/Demographic\_Reports/documents/2020-1850\_STCO\_IncCities.xlsx

# This original file has multiple header lines. It is historical, so can be considered static.

# Clean it externally, save it, and read it in for to merge to SFdata:

SFCityPops <- read\_csv("C:\\Data Science CAS\\IST 687\\Team 2\\2020-1850\_STCO\_IncCities for IST687.csv")

str(SFCityPops)

head(SFCityPops,25)

summary(SFCityPops)

print(SFCityPops,n=500)

# drop `County` - already have it in SFdata:

SFCityPops <- SFCityPops[,-1]

# Get column name of city to match the name in SFdata, and give the population columns 'plain English` names:

colnames(SFCityPops) <- c("city","Population\_2000","Population\_2001","Population\_2020")

# Before merging, check that the values of city in SFdata and SFCityPops match.

table(SFdata$city)

table(SFCityPops$city)

# We need to make SFCityPops$city lowercase:

SFCityPops$city <- tolower(SFCityPops$city)

# directional left/outer join merge using `city.` Keeps all values/rows in SFdata whether or not the city is in SFCityPops.

# https://www.rdocumentation.org/packages/base/versions/3.6.2/topics/merge

SFdata <- merge(SFdata, SFCityPops, by = "city", all.x = TRUE )

# any cities with no pop data?

tapply(SFdata$Population\_2020,SFdata$city,is.na)

# Yes - ben lomond, boulder cr, cambrian, corralitos, east bay, el sobrante, felton, greenbrae, guerneville,

# kentfield, marin, montara, napa county, north bay, peninsula, pescadero, redwood shores, russian river,

# sf bay area, soquel, south bay, woodacre

# Look them up on Wikipedia- has 2020 population only. Hard-code to assign populations.

SFdata$Population\_2020 <- ifelse(SFdata$city == "ben lomond", 6337, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "boulder cr", 5305, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "cambrian", 3015, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "corralitos", 2260, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "el sobrante", 14779, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "felton", 3197, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "greenbrae", 7423, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "guerneville", 4747, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "kentfield", 7423, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "marin", 260206, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "montara", 2833, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "napa county", 136207, SFdata$Population\_2020)

# No data

# SFdata$Population\_2020 <- ifelse(SFdata$city == "north bay", SFdata$Population\_2020, SFdata$Population\_2020)

# No data

# SFdata$Population\_2020 <- ifelse(SFdata$city == "peninsula", SFdata$Population\_2020, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "pescadero", 595, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "redwood shores", 4076, SFdata$Population\_2020)

# No data

# SFdata$Population\_2020 <- ifelse(SFdata$city == "russian river", SFdata$Population\_2020, SFdata$Population\_2020)

# No data

# SFdata$Population\_2020 <- ifelse(SFdata$city == "sf bay area", SFdata$Population\_2020, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "soquel", 10721, SFdata$Population\_2020)

# No data

# SFdata$Population\_2020 <- ifelse(SFdata$city == "south bay", SFdata$Population\_2020, SFdata$Population\_2020)

SFdata$Population\_2020 <- ifelse(SFdata$city == "woodacre", 1464, SFdata$Population\_2020)

# Check that most cities (except those noted above) have no NAs (i.e., are FALSE for is.na) on Population\_2020:

tapply(SFdata$Population\_2020,SFdata$city,is.na)

# There are NAs for Population\_2000, Population\_2001, and Population\_2020. Get imputed versions for regression.

summary(SFdata$Population\_2000)

summary(SFdata$Population\_2001)

summary(SFdata$Population\_2020)

SFdata$Population\_2000\_i <- na\_interpolation(SFdata$Population\_2000)

SFdata$Population\_2001\_i <- na\_interpolation(SFdata$Population\_2001)

SFdata$Population\_2020\_i <- na\_interpolation(SFdata$Population\_2020)

summary(SFdata$Population\_2000\_i)

summary(SFdata$Population\_2001\_i)

summary(SFdata$Population\_2020\_i)

1. Code for adding mean city population to each record:

# To prevent issues with processing regression output datasets,

# where names of variables may derive from county/city/nhood

# (e.g., for feature selection like backwards elimination):

# convert `/` and spaces to underscores. Underscores behave

# much better as variables names than `/` and spaces.

# Save these geographies as new columns.

#table(SFdata$county)

SFdata$county\_cln <- str\_replace\_all(SFdata$county, " ", "\_")

#table(SFdata$county\_cln)

#table(SFdata$city)

SFdata$city\_cln <- str\_replace\_all(SFdata$city, " ", "\_")

#table(SFdata$city\_cln)

# For nhood, convert both slashes and spaces.

#table(SFdata$nhood)

SFdata$nhood\_cln <- str\_replace\_all(SFdata$nhood, " / ", "\_")

SFdata$nhood\_cln <- str\_replace\_all(SFdata$nhood\_cln, " ", "\_")

#table(SFdata$nhood\_cln)

1. Code for running the initial full-dataset model and the backwards elimination routine. The full model, and the variables for removal as per the backwards elimination routine, are embedded in the code. Variables identified for removal from the backwards elimination process are removed from the dataframe, and the linear model re-run to obtain the final, full-dataset model.

# Get initial full model:

m\_full\_nhood <- lm(price ~ ., data = SFdata\_lm\_vars)

summary(m\_full\_nhood)

lm(formula = price ~ ., data = SFdata\_lm\_vars)

Residuals:

Min 1Q Median 3Q Max

-6262.7 -469.2 -50.5 353.9 18790.5

Coefficients:

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

(Intercept) -2.288e+05 3.949e+03 -57.939 < 2e-16 \*\*\*

year 1.138e+02 1.960e+00 58.042 < 2e-16 \*\*\*

beds 7.381e+02 8.749e+00 84.356 < 2e-16 \*\*\*

baths\_na\_interpolation\_i 1.016e+02 1.481e+01 6.858 7.26e-12 \*\*\*

sqft\_na\_interpolation\_i 2.710e-01 1.278e-02 21.199 < 2e-16 \*\*\*

nhood\_cln\_alameda -1.159e+02 1.333e+02 -0.869 0.384726

nhood\_cln\_alamo\_square 8.202e+02 2.250e+02 3.646 0.000268 \*\*\*

nhood\_cln\_albany\_el\_cerrito 8.789e+01 1.113e+02 0.790 0.429831

nhood\_cln\_almaden 7.255e+01 5.969e+02 0.122 0.903268

nhood\_cln\_bayview 1.260e+02 1.579e+02 0.798 0.425135

nhood\_cln\_belmont\_san\_carlos 3.641e+02 9.323e+01 3.905 9.45e-05 \*\*\*

nhood\_cln\_ben\_lomond -3.946e+02 3.468e+02 -1.138 0.255303

nhood\_cln\_berkeley 4.726e+02 7.622e+01 6.200 5.79e-10 \*\*\*

nhood\_cln\_berkeley\_downtown 2.817e+02 3.676e+02 0.766 0.443455

nhood\_cln\_berkeley\_north\_hills 6.478e+02 1.316e+02 4.924 8.57e-07 \*\*\*

nhood\_cln\_berkeley\_south 4.928e+02 5.970e+02 0.825 0.409160

nhood\_cln\_berkeley\_west 5.333e+02 5.177e+02 1.030 0.302895

nhood\_cln\_bernal 1.069e+03 1.595e+02 6.700 2.17e-11 \*\*\*

nhood\_cln\_brentwood\_oakley\_discovery\_bay -1.695e+03 1.182e+02 -14.341 < 2e-16 \*\*\*

nhood\_cln\_brisbane 3.754e+02 2.900e+02 1.295 0.195511

nhood\_cln\_burlingame\_hillsborough 7.688e+02 1.058e+02 7.266 3.91e-13 \*\*\*

nhood\_cln\_campbell 5.820e+01 1.086e+02 0.536 0.591966

nhood\_cln\_candlestick\_point 1.316e+03 5.968e+02 2.205 0.027459 \*

nhood\_cln\_capitola\_soquel -7.087e+01 1.308e+02 -0.542 0.587874

nhood\_cln\_castro 1.407e+03 1.139e+02 12.354 < 2e-16 \*\*\*

nhood\_cln\_CCSF 6.099e+02 1.257e+02 4.853 1.23e-06 \*\*\*

nhood\_cln\_civic\_van\_ness 1.179e+03 1.015e+02 11.617 < 2e-16 \*\*\*

nhood\_cln\_clayton -2.312e+02 7.304e+02 -0.317 0.751562

nhood\_cln\_cole\_valley 1.559e+03 1.720e+02 9.067 < 2e-16 \*\*\*

nhood\_cln\_concord\_pleasant\_hill\_martinez -4.987e+02 8.687e+01 -5.742 9.58e-09 \*\*\*

nhood\_cln\_corralitos -8.236e+01 1.201e+02 -0.686 0.492783

nhood\_cln\_corte\_madera 3.524e+02 1.785e+02 1.974 0.048381 \*

nhood\_cln\_cupertino 3.517e+02 8.686e+01 4.049 5.17e-05 \*\*\*

nhood\_cln\_daly\_city 4.319e+01 9.648e+01 0.448 0.654424

nhood\_cln\_danville\_san\_ramon -1.659e+00 1.046e+02 -0.016 0.987350

nhood\_cln\_diamond\_heights 1.240e+03 2.413e+02 5.139 2.79e-07 \*\*\*

nhood\_cln\_downtown 1.716e+03 2.477e+02 6.927 4.48e-12 \*\*\*

nhood\_cln\_dublin\_pleasanton -3.165e+02 8.533e+01 -3.709 0.000209 \*\*\*

nhood\_cln\_emeryville 4.766e+02 1.264e+02 3.772 0.000163 \*\*\*

nhood\_cln\_excelsior\_outer\_mission 1.714e+02 1.292e+02 1.326 0.184849

nhood\_cln\_fairfield\_vacaville -1.063e+03 1.010e+02 -10.529 < 2e-16 \*\*\*

nhood\_cln\_financial\_district 2.540e+03 1.264e+02 20.101 < 2e-16 \*\*\*

nhood\_cln\_foster\_city 5.069e+02 1.130e+02 4.484 7.37e-06 \*\*\*

nhood\_cln\_gilroy -5.303e+02 1.536e+02 -3.451 0.000560 \*\*\*

nhood\_cln\_glen\_park 1.294e+03 2.300e+02 5.625 1.89e-08 \*\*\*

nhood\_cln\_greenbrae 5.722e+02 2.548e+02 2.246 0.024710 \*

nhood\_cln\_haight\_ashbury 1.150e+03 1.646e+02 6.990 2.88e-12 \*\*\*

nhood\_cln\_hayes\_valley 1.437e+03 1.387e+02 10.362 < 2e-16 \*\*\*

nhood\_cln\_hayward\_castro\_valley -4.030e+02 8.885e+01 -4.535 5.81e-06 \*\*\*

nhood\_cln\_healdsburg\_windsor -6.411e+02 1.596e+02 -4.018 5.91e-05 \*\*\*

nhood\_cln\_hercules\_pinole\_san\_pablo\_el\_sob -6.002e+02 1.139e+02 -5.268 1.40e-07 \*\*\*

nhood\_cln\_hunters\_point 1.060e+03 7.301e+02 1.452 0.146569

nhood\_cln\_ingleside 2.561e+02 1.644e+02 1.557 0.119481

nhood\_cln\_inner\_richmond 8.105e+02 1.084e+02 7.474 8.25e-14 \*\*\*

nhood\_cln\_inner\_sunset 7.755e+02 1.000e+02 7.751 9.73e-15 \*\*\*

nhood\_cln\_lafayette\_orinda\_moraga 3.055e+02 1.485e+02 2.058 0.039635 \*

nhood\_cln\_lakeshore 5.339e+01 2.477e+02 0.216 0.829334

nhood\_cln\_larkspur 9.275e+02 1.484e+02 6.252 4.18e-10 \*\*\*

nhood\_cln\_los\_altos 1.606e+03 1.344e+02 11.951 < 2e-16 \*\*\*

nhood\_cln\_los\_gatos 3.737e+02 1.224e+02 3.053 0.002273 \*\*

nhood\_cln\_lower\_haight 1.271e+03 2.079e+02 6.112 1.01e-09 \*\*\*

nhood\_cln\_lower\_pac\_hts 1.457e+03 1.100e+02 13.247 < 2e-16 \*\*\*

nhood\_cln\_marin 2.993e+02 1.914e+02 1.563 0.117975

nhood\_cln\_marina\_cow\_hollow 2.120e+03 8.711e+01 24.337 < 2e-16 \*\*\*

nhood\_cln\_menlo\_park 8.703e+02 1.150e+02 7.568 4.03e-14 \*\*\*

nhood\_cln\_mill\_valley 1.094e+03 1.087e+02 10.062 < 2e-16 \*\*\*

nhood\_cln\_millbrae 3.541e+02 1.721e+02 2.058 0.039614 \*

nhood\_cln\_milpitas -6.982e+01 1.250e+02 -0.559 0.576512

nhood\_cln\_mission\_district 1.118e+03 8.850e+01 12.631 < 2e-16 \*\*\*

nhood\_cln\_mountain\_view 4.696e+02 7.605e+01 6.175 6.81e-10 \*\*\*

nhood\_cln\_napa\_county -3.210e+02 1.074e+02 -2.989 0.002802 \*\*

nhood\_cln\_nob\_hill 1.461e+03 7.858e+01 18.590 < 2e-16 \*\*\*

nhood\_cln\_noe\_valley 1.659e+03 1.077e+02 15.406 < 2e-16 \*\*\*

nhood\_cln\_NOPA 1.160e+03 1.471e+02 7.886 3.34e-15 \*\*\*

nhood\_cln\_north\_beach\_telegraph\_hill 1.785e+03 1.190e+02 15.002 < 2e-16 \*\*\*

nhood\_cln\_novato 6.971e+01 1.111e+02 0.628 0.530319

nhood\_cln\_oakland -1.556e+02 3.014e+02 -0.516 0.605670

nhood\_cln\_oakland\_east -6.340e+02 9.465e+01 -6.698 2.19e-11 \*\*\*

nhood\_cln\_oakland\_hills\_north\_montclair 5.099e+02 9.925e+01 5.138 2.82e-07 \*\*\*

nhood\_cln\_oakland\_hills\_south 8.982e+00 1.181e+02 0.076 0.939359

nhood\_cln\_oakland\_lake\_merritt\_downtown 1.965e+02 7.457e+01 2.635 0.008412 \*\*

nhood\_cln\_oakland\_north 1.230e+02 1.388e+02 0.886 0.375419

nhood\_cln\_oakland\_west -3.881e+02 1.138e+02 -3.409 0.000653 \*\*\*

nhood\_cln\_outer\_richmond 6.722e+02 2.705e+02 2.485 0.012962 \*

nhood\_cln\_outer\_sunset 3.704e+02 1.045e+02 3.544 0.000395 \*\*\*

nhood\_cln\_pacific\_heights 2.654e+03 8.030e+01 33.051 < 2e-16 \*\*\*

nhood\_cln\_pacifica 3.467e+02 1.153e+02 3.006 0.002652 \*\*

nhood\_cln\_palo\_alto 1.132e+03 7.961e+01 14.221 < 2e-16 \*\*\*

nhood\_cln\_parkside 5.043e+02 1.117e+02 4.516 6.34e-06 \*\*\*

nhood\_cln\_petaluma -2.678e+02 9.892e+01 -2.707 0.006803 \*\*

nhood\_cln\_pittsburg\_antioch -1.187e+03 8.360e+01 -14.197 < 2e-16 \*\*\*

nhood\_cln\_portola 6.928e+02 2.159e+02 3.209 0.001334 \*\*

nhood\_cln\_potrero\_hill 1.624e+03 1.237e+02 13.129 < 2e-16 \*\*\*

nhood\_cln\_presidio\_hts\_laurel\_hts\_lake\_st 2.001e+03 1.332e+02 15.020 < 2e-16 \*\*\*

nhood\_cln\_redwood\_city 5.065e+02 9.174e+01 5.521 3.43e-08 \*\*\*

nhood\_cln\_redwood\_shores 7.191e+02 2.252e+02 3.194 0.001406 \*\*

nhood\_cln\_richmond\_point\_annex -5.927e+02 1.168e+02 -5.074 3.95e-07 \*\*\*

nhood\_cln\_rohnert\_pk\_cotati -5.941e+02 1.219e+02 -4.874 1.11e-06 \*\*\*

nhood\_cln\_russian\_hill 2.425e+03 9.507e+01 25.505 < 2e-16 \*\*\*

nhood\_cln\_russian\_river -3.711e+02 2.008e+02 -1.848 0.064607 .

nhood\_cln\_san\_anselmo 4.259e+02 1.945e+02 2.190 0.028538 \*

nhood\_cln\_san\_bruno 2.733e+02 1.136e+02 2.406 0.016136 \*

nhood\_cln\_san\_francisco 7.772e+01 1.009e+02 0.770 0.441070

nhood\_cln\_san\_jose -3.516e+02 2.159e+02 -1.629 0.103386

nhood\_cln\_san\_jose\_central 4.723e+01 9.086e+01 0.520 0.603194

nhood\_cln\_san\_jose\_east -4.288e+02 1.164e+02 -3.683 0.000231 \*\*\*

nhood\_cln\_san\_jose\_north -6.941e+00 9.891e+01 -0.070 0.944057

nhood\_cln\_san\_jose\_south -1.792e+02 7.929e+01 -2.260 0.023806 \*

nhood\_cln\_san\_jose\_west 8.561e+01 8.119e+01 1.054 0.291690

nhood\_cln\_san\_leandro -4.111e+02 1.104e+02 -3.723 0.000198 \*\*\*

nhood\_cln\_san\_mateo 4.852e+02 7.831e+01 6.196 5.94e-10 \*\*\*

nhood\_cln\_san\_rafael 4.085e+02 9.069e+01 4.505 6.70e-06 \*\*\*

nhood\_cln\_santa\_clara 1.459e+02 7.546e+01 1.934 0.053106 .

nhood\_cln\_santa\_rosa -4.670e+02 7.293e+01 -6.404 1.56e-10 \*\*\*

nhood\_cln\_saratoga 1.368e+03 1.686e+02 8.117 5.17e-16 \*\*\*

nhood\_cln\_sausalito 1.241e+03 1.246e+02 9.965 < 2e-16 \*\*\*

nhood\_cln\_sea\_cliff 6.257e+02 9.398e+01 6.657 2.90e-11 \*\*\*

nhood\_cln\_SOMA\_south\_beach 2.321e+03 6.912e+01 33.587 < 2e-16 \*\*\*

nhood\_cln\_sonoma 4.544e+01 1.663e+02 0.273 0.784687

nhood\_cln\_sunnyvale 2.195e+02 7.721e+01 2.843 0.004476 \*\*

nhood\_cln\_tenderloin 5.668e+02 1.668e+02 3.398 0.000681 \*\*\*

nhood\_cln\_tiburon\_belvedere 2.635e+03 1.510e+02 17.454 < 2e-16 \*\*\*

nhood\_cln\_twin\_peaks 1.586e+03 1.974e+02 8.037 9.93e-16 \*\*\*

nhood\_cln\_union\_city -2.171e+02 7.757e+01 -2.799 0.005131 \*\*

nhood\_cln\_USF\_anza\_vista 7.506e+02 1.974e+02 3.801 0.000144 \*\*\*

nhood\_cln\_vallejo\_benicia -9.456e+02 9.077e+01 -10.417 < 2e-16 \*\*\*

nhood\_cln\_visitacion\_valley 1.859e+02 2.355e+02 0.790 0.429716

nhood\_cln\_walnut\_creek 9.812e+01 9.864e+01 0.995 0.319852

nhood\_cln\_west\_portal\_forest\_hills 2.021e+03 1.726e+02 11.711 < 2e-16 \*\*\*

nhood\_cln\_western\_addition 1.023e+03 2.008e+02 5.094 3.55e-07 \*\*\*

nhood\_cln\_willow\_glen\_cambrian 5.038e+01 1.207e+02 0.418 0.676293

nhood\_cln\_woodside 7.304e+02 2.356e+02 3.100 0.001937 \*\*

---

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

Residual standard error: 1030 on 13971 degrees of freedom

Multiple R-squared: 0.5933, Adjusted R-squared: 0.5895

F-statistic: 155.6 on 131 and 13971 DF, p-value: < 2.2e-16l full model:

# Now apply the backwards elimination routine to ultimately retain significant predictors:

# Eliminate predictors not significant at the 0.05 level.

m\_full\_nhood\_step\_backward <- ols\_step\_backward\_p(m\_full\_nhood, prem = 0.05)

# This gives all the details-- each step:

# m\_full\_nhood\_step\_backward <- ols\_step\_backward\_p(m\_full\_nhood, details = TRUE)

# These inputs were eliminated. Remainder of predictors is the model.

M\_full\_nhood\_step\_backward

Elimination Summary

---------------------------------------------------------------------------------------------------------

Variable Adj.

Step Removed R-Square R-Square C(p) AIC RMSE

---------------------------------------------------------------------------------------------------------

1 nhood\_cln\_danville\_san\_ramon 0.5933 0.5895 130.0003 235831.8914 1030.1111

2 nhood\_cln\_san\_jose\_north 0.5933 0.5896 128.0049 235829.8961 1030.0744

3 nhood\_cln\_oakland\_hills\_south 0.5933 0.5896 126.0136 235827.9049 1030.0379

4 nhood\_cln\_almaden 0.5933 0.5896 124.0286 235825.9200 1030.0016

5 nhood\_cln\_lakeshore 0.5933 0.5897 122.0762 235823.9680 1029.9665

6 nhood\_cln\_sonoma 0.5933 0.5897 120.1506 235822.0432 1029.9324

7 nhood\_cln\_clayton 0.5933 0.5897 118.2539 235820.1474 1029.8993

8 nhood\_cln\_willow\_glen\_cambrian 0.5933 0.5897 116.4243 235818.3194 1029.8688

9 nhood\_cln\_daly\_city 0.5933 0.5898 114.5877 235816.4844 1029.8380

10 nhood\_cln\_san\_jose\_central 0.5933 0.5898 112.7698 235814.6681 1029.8079

11 nhood\_cln\_campbell 0.5933 0.5898 110.9364 235812.8364 1029.7772

12 nhood\_cln\_novato 0.5933 0.5898 109.1669 235811.0690 1029.7488

13 nhood\_cln\_san\_francisco 0.5933 0.5899 107.5102 235809.4155 1029.7247

14 nhood\_cln\_albany\_el\_cerrito 0.5933 0.5899 105.8441 235807.7525 1029.7002

15 nhood\_cln\_bayview 0.5933 0.5899 104.2280 235806.1400 1029.6775

16 nhood\_cln\_oakland 0.5933 0.5899 102.6331 235804.5488 1029.6556

17 nhood\_cln\_visitacion\_valley 0.5932 0.5899 101.0716 235802.9914 1029.6350

18 nhood\_cln\_oakland\_north 0.5932 0.5899 99.5179 235801.4418 1029.6146

19 nhood\_cln\_berkeley\_downtown 0.5932 0.590 97.9634 235799.8915 1029.5942

20 nhood\_cln\_walnut\_creek 0.5932 0.590 96.4115 235798.3437 1029.5739

21 nhood\_cln\_san\_jose\_west 0.5932 0.590 94.7979 235796.7335 1029.5514

22 nhood\_cln\_berkeley\_south 0.5932 0.590 93.3544 235795.2951 1029.5351

23 nhood\_cln\_berkeley\_west 0.5931 0.590 92.2353 235794.1839 1029.5307

24 nhood\_cln\_capitola\_soquel 0.5931 0.590 91.1709 235793.1279 1029.5284

25 nhood\_cln\_milpitas 0.5931 0.590 90.1260 235792.0915 1029.5268

26 nhood\_cln\_excelsior\_outer\_mission 0.5931 0.590 89.2735 235791.2491 1029.5323

27 nhood\_cln\_corralitos 0.593 0.590 88.5620 235790.5489 1029.5429

28 nhood\_cln\_brisbane 0.593 0.590 87.9114 235789.9100 1029.5558

29 nhood\_cln\_alameda 0.5929 0.590 87.5169 235789.5291 1029.5782

30 nhood\_cln\_ben\_lomond 0.5929 0.590 87.0982 235789.1238 1029.5996

31 nhood\_cln\_ingleside 0.5928 0.5899 87.0073 235789.0487 1029.6331

32 nhood\_cln\_marin 0.5928 0.5899 86.9250 235788.9820 1029.6669

33 nhood\_cln\_hunters\_point 0.5927 0.5899 86.8753 235788.9479 1029.7019

34 nhood\_cln\_santa\_clara 0.5926 0.5898 87.6500 235789.7444 1029.7672

35 nhood\_cln\_corte\_madera 0.5926 0.5898 88.6103 235790.7273 1029.8394

36 nhood\_cln\_lafayette\_orinda\_moraga 0.5925 0.5897 89.6754 235791.8151 1029.9154

37 nhood\_cln\_millbrae 0.5924 0.5896 90.7551 235792.9171 1029.9919

38 nhood\_cln\_san\_anselmo 0.5923 0.5896 92.3376 235794.5244 1030.0868

39 nhood\_cln\_san\_bruno 0.5922 0.5895 94.1287 235796.3409 1030.1895

# Get final-model dataframe - Remove eliminated (non-significant) predictors:

SFdata\_lm\_vars\_final <- subset(SFdata\_lm\_vars, select = - c(

nhood\_cln\_capitola\_soquel,

nhood\_cln\_san\_jose\_central,

nhood\_cln\_woodside,

nhood\_cln\_willow\_glen\_cambrian,

nhood\_cln\_san\_jose\_north,

nhood\_cln\_alameda,

nhood\_cln\_almaden,

nhood\_cln\_novato,

nhood\_cln\_oakland,

nhood\_cln\_milpitas,

nhood\_cln\_visitacion\_valley,

nhood\_cln\_lakeshore,

nhood\_cln\_daly\_city,

nhood\_cln\_bayview,

nhood\_cln\_oakland\_hills\_south,

nhood\_cln\_corte\_madera,

nhood\_cln\_san\_francisco,

nhood\_cln\_san\_jose\_west,

nhood\_cln\_campbell,

nhood\_cln\_oakland\_north,

nhood\_cln\_excelsior\_outer\_mission,

nhood\_cln\_ingleside,

nhood\_cln\_berkeley\_south,

nhood\_cln\_lafayette\_orinda\_moraga,

nhood\_cln\_berkeley\_west,

nhood\_cln\_walnut\_creek,

nhood\_cln\_berkeley\_downtown,

nhood\_cln\_hunters\_point,

nhood\_cln\_ben\_lomond,

nhood\_cln\_san\_jose,

nhood\_cln\_albany\_el\_cerrito,

nhood\_cln\_santa\_clara,

nhood\_cln\_millbrae,

nhood\_cln\_san\_rafael,

nhood\_cln\_outer\_richmond,

nhood\_cln\_candlestick\_point

))

# check to see the above nonsignificant predictors are no longer present.

# str(SFdata\_lm\_vars\_final)

# Run final model. Save it so we can get predicted values for new observations, etc.

m\_full\_nhood\_final <- lm(price ~ ., data = SFdata\_lm\_vars\_final)

str(m\_full\_nhood\_final)

summary(m\_full\_nhood\_final)

1. Code for running the 75% training/25% validation linear model. The process is similar to that for the full-dataset model, except that the model is developed with the 75% training dataset. Then, validation dataset cases are run through the model developed through the training dataset, and root mean square residual is assessed:

# Run a training/validation analysis of the regression. Do a 75%/25% random split.

# After training, run the test set through the model and get the RMSE.

# Note: this is a little tricky, as the training set may not have neighborhoods that

# are in the test set, and in this way the training model may `under-represent`

# such neighborhoods. We will proceed, noting this as a possible caveat.

library(caret)

set.seed(313)

trainList <- createDataPartition(y=SFdata\_lm\_vars\_final$price, p=.75, list=FALSE)

# For fitting the model.

trainSet <- SFdata\_lm\_vars\_final[trainList,]

# For validating `new` cases.

testSet <- SFdata\_lm\_vars\_final[-trainList,]

# See that some records are not kept in the training set:

head(trainList,25)

# check the 75/25 split.

dim(trainSet)

dim(testSet)

# Now fit the lm on the training set.

m\_full\_nhood\_train <- lm(price ~ ., data = trainSet)

# We see some non-significant predictors:

summary(m\_full\_nhood\_train)

# Trim the model by applying the backwards elimination routine:

m\_full\_nhood\_train\_step\_backward <- ols\_step\_backward\_p(m\_full\_nhood\_train, prem = 0.05)

# These inputs were eliminated. Remainder of predictors will be the final training model.

m\_full\_nhood\_train\_step\_backward

# Get final-model dataframe - Remove eliminated (non-significant) predictors:

trainSet\_final <- subset(trainSet, select = - c(

nhood\_cln\_clayton,

nhood\_cln\_brisbane,

nhood\_cln\_sonoma,

nhood\_cln\_marin,

nhood\_cln\_danville\_san\_ramon,

nhood\_cln\_corralitos,

nhood\_cln\_greenbrae,

nhood\_cln\_oakland\_lake\_merritt\_downtown,

nhood\_cln\_san\_bruno,

nhood\_cln\_san\_anselmo,

nhood\_cln\_diamond\_heights,

nhood\_cln\_pacifica,

nhood\_cln\_sunnyvale

))

# Run final model. Save it so we can get predicted values for new observations, etc.

m\_full\_nhood\_final\_trainSet <- lm(price ~ ., data = trainSet\_final)

str(m\_full\_nhood\_final\_trainSet)

summary(m\_full\_nhood\_final\_trainSet)

Call:

lm(formula = price ~ ., data = trainSet\_final)

Residuals:

Min 1Q Median 3Q Max

-6151.9 -469.1 -43.1 362.6 18643.4

Coefficients:

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

(Intercept) -2.252e+05 4.486e+03 -50.197 < 2e-16 \*\*\*

year 1.121e+02 2.228e+00 50.286 < 2e-16 \*\*\*

beds 7.253e+02 9.880e+00 73.406 < 2e-16 \*\*\*

baths\_na\_interpolation\_i 8.421e+01 1.691e+01 4.980 6.46e-07 \*\*\*

sqft\_na\_interpolation\_i 3.007e-01 1.541e-02 19.515 < 2e-16 \*\*\*

nhood\_cln\_alamo\_square 5.889e+02 2.484e+02 2.371 0.017779 \*

nhood\_cln\_belmont\_san\_carlos 2.734e+02 9.354e+01 2.923 0.003473 \*\*

nhood\_cln\_berkeley 3.257e+02 6.846e+01 4.757 1.99e-06 \*\*\*

nhood\_cln\_berkeley\_north\_hills 5.286e+02 1.261e+02 4.193 2.78e-05 \*\*\*

nhood\_cln\_bernal 9.864e+02 1.646e+02 5.994 2.11e-09 \*\*\*

nhood\_cln\_brentwood\_oakley\_discovery\_bay -1.776e+03 1.217e+02 -14.589 < 2e-16 \*\*\*

nhood\_cln\_burlingame\_hillsborough 6.477e+02 1.070e+02 6.051 1.49e-09 \*\*\*

nhood\_cln\_castro 1.246e+03 1.226e+02 10.167 < 2e-16 \*\*\*

nhood\_cln\_CCSF 5.620e+02 1.321e+02 4.254 2.11e-05 \*\*\*

nhood\_cln\_civic\_van\_ness 1.035e+03 1.028e+02 10.068 < 2e-16 \*\*\*

nhood\_cln\_cole\_valley 1.445e+03 1.815e+02 7.964 1.84e-15 \*\*\*

nhood\_cln\_concord\_pleasant\_hill\_martinez -6.100e+02 8.852e+01 -6.891 5.86e-12 \*\*\*

nhood\_cln\_cupertino 2.557e+02 8.534e+01 2.996 0.002739 \*\*

nhood\_cln\_downtown 1.318e+03 2.645e+02 4.982 6.38e-07 \*\*\*

nhood\_cln\_dublin\_pleasanton -3.664e+02 8.346e+01 -4.390 1.14e-05 \*\*\*

nhood\_cln\_emeryville 4.042e+02 1.310e+02 3.086 0.002036 \*\*

nhood\_cln\_fairfield\_vacaville -1.097e+03 1.034e+02 -10.607 < 2e-16 \*\*\*

nhood\_cln\_financial\_district 2.090e+03 1.332e+02 15.691 < 2e-16 \*\*\*

nhood\_cln\_foster\_city 4.008e+02 1.186e+02 3.381 0.000724 \*\*\*

nhood\_cln\_gilroy -6.215e+02 1.690e+02 -3.677 0.000237 \*\*\*

nhood\_cln\_glen\_park 1.149e+03 2.736e+02 4.200 2.69e-05 \*\*\*

nhood\_cln\_haight\_ashbury 9.699e+02 1.939e+02 5.002 5.77e-07 \*\*\*

nhood\_cln\_hayes\_valley 1.331e+03 1.569e+02 8.481 < 2e-16 \*\*\*

nhood\_cln\_hayward\_castro\_valley -5.319e+02 8.376e+01 -6.351 2.23e-10 \*\*\*

nhood\_cln\_healdsburg\_windsor -7.809e+02 1.713e+02 -4.558 5.23e-06 \*\*\*

nhood\_cln\_hercules\_pinole\_san\_pablo\_el\_sob -6.940e+02 1.178e+02 -5.889 4.00e-09 \*\*\*

nhood\_cln\_inner\_richmond 6.770e+02 1.149e+02 5.890 3.97e-09 \*\*\*

nhood\_cln\_inner\_sunset 6.279e+02 1.009e+02 6.225 5.01e-10 \*\*\*

nhood\_cln\_larkspur 7.992e+02 1.605e+02 4.978 6.53e-07 \*\*\*

nhood\_cln\_los\_altos 1.400e+03 1.519e+02 9.213 < 2e-16 \*\*\*

nhood\_cln\_los\_gatos 2.940e+02 1.342e+02 2.190 0.028552 \*

nhood\_cln\_lower\_haight 1.060e+03 2.186e+02 4.851 1.25e-06 \*\*\*

nhood\_cln\_lower\_pac\_hts 1.318e+03 1.130e+02 11.664 < 2e-16 \*\*\*

nhood\_cln\_marina\_cow\_hollow 1.939e+03 8.513e+01 22.778 < 2e-16 \*\*\*

nhood\_cln\_menlo\_park 7.226e+02 1.171e+02 6.171 7.02e-10 \*\*\*

nhood\_cln\_mill\_valley 9.566e+02 1.124e+02 8.512 < 2e-16 \*\*\*

nhood\_cln\_mission\_district 9.189e+02 8.676e+01 10.591 < 2e-16 \*\*\*

nhood\_cln\_mountain\_view 3.524e+02 6.963e+01 5.062 4.23e-07 \*\*\*

nhood\_cln\_napa\_county -6.142e+02 1.064e+02 -5.770 8.14e-09 \*\*\*

nhood\_cln\_nob\_hill 1.269e+03 7.215e+01 17.581 < 2e-16 \*\*\*

nhood\_cln\_noe\_valley 1.463e+03 1.170e+02 12.503 < 2e-16 \*\*\*

nhood\_cln\_NOPA 1.026e+03 1.569e+02 6.539 6.46e-11 \*\*\*

nhood\_cln\_north\_beach\_telegraph\_hill 1.743e+03 1.218e+02 14.308 < 2e-16 \*\*\*

nhood\_cln\_oakland\_east -7.734e+02 9.289e+01 -8.327 < 2e-16 \*\*\*

nhood\_cln\_oakland\_hills\_north\_montclair 3.531e+02 9.881e+01 3.573 0.000354 \*\*\*

nhood\_cln\_oakland\_west -4.860e+02 1.287e+02 -3.776 0.000160 \*\*\*

nhood\_cln\_outer\_sunset 2.926e+02 1.069e+02 2.737 0.006207 \*\*

nhood\_cln\_pacific\_heights 2.519e+03 7.607e+01 33.119 < 2e-16 \*\*\*

nhood\_cln\_palo\_alto 9.432e+02 7.432e+01 12.691 < 2e-16 \*\*\*

nhood\_cln\_parkside 3.763e+02 1.135e+02 3.314 0.000922 \*\*\*

nhood\_cln\_petaluma -4.203e+02 9.994e+01 -4.205 2.63e-05 \*\*\*

nhood\_cln\_pittsburg\_antioch -1.329e+03 8.128e+01 -16.356 < 2e-16 \*\*\*

nhood\_cln\_portola 5.234e+02 2.237e+02 2.339 0.019338 \*

nhood\_cln\_potrero\_hill 1.506e+03 1.261e+02 11.945 < 2e-16 \*\*\*

nhood\_cln\_presidio\_hts\_laurel\_hts\_lake\_st 2.020e+03 1.456e+02 13.869 < 2e-16 \*\*\*

nhood\_cln\_redwood\_city 3.358e+02 8.925e+01 3.762 0.000170 \*\*\*

nhood\_cln\_redwood\_shores 6.336e+02 2.416e+02 2.623 0.008733 \*\*

nhood\_cln\_richmond\_point\_annex -6.983e+02 1.193e+02 -5.851 5.02e-09 \*\*\*

nhood\_cln\_rohnert\_pk\_cotati -7.104e+02 1.320e+02 -5.380 7.60e-08 \*\*\*

nhood\_cln\_russian\_hill 2.421e+03 9.575e+01 25.280 < 2e-16 \*\*\*

nhood\_cln\_russian\_river -6.116e+02 2.484e+02 -2.462 0.013839 \*

nhood\_cln\_san\_jose\_east -5.097e+02 1.311e+02 -3.888 0.000102 \*\*\*

nhood\_cln\_san\_jose\_south -3.042e+02 7.394e+01 -4.114 3.93e-05 \*\*\*

nhood\_cln\_san\_leandro -5.613e+02 1.105e+02 -5.082 3.81e-07 \*\*\*

nhood\_cln\_san\_mateo 3.763e+02 7.260e+01 5.183 2.22e-07 \*\*\*

nhood\_cln\_santa\_rosa -5.836e+02 6.540e+01 -8.924 < 2e-16 \*\*\*

nhood\_cln\_saratoga 1.514e+03 1.878e+02 8.060 8.46e-16 \*\*\*

nhood\_cln\_sausalito 1.146e+03 1.261e+02 9.082 < 2e-16 \*\*\*

nhood\_cln\_sea\_cliff 4.847e+02 9.310e+01 5.206 1.97e-07 \*\*\*

nhood\_cln\_SOMA\_south\_beach 2.240e+03 5.996e+01 37.359 < 2e-16 \*\*\*

nhood\_cln\_tenderloin 4.093e+02 1.796e+02 2.279 0.022692 \*

nhood\_cln\_tiburon\_belvedere 2.556e+03 1.689e+02 15.133 < 2e-16 \*\*\*

nhood\_cln\_twin\_peaks 8.475e+02 2.237e+02 3.789 0.000152 \*\*\*

nhood\_cln\_union\_city -3.647e+02 7.207e+01 -5.061 4.24e-07 \*\*\*

nhood\_cln\_USF\_anza\_vista 6.020e+02 2.186e+02 2.754 0.005893 \*\*

nhood\_cln\_vallejo\_benicia -1.061e+03 8.953e+01 -11.853 < 2e-16 \*\*\*

nhood\_cln\_west\_portal\_forest\_hills 1.977e+03 1.795e+02 11.010 < 2e-16 \*\*\*

nhood\_cln\_western\_addition 8.885e+02 2.293e+02 3.875 0.000107 \*\*\*

---

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

Residual standard error: 1021 on 10496 degrees of freedom

Multiple R-squared: 0.5928, Adjusted R-squared: 0.5896

F-statistic: 186.3 on 82 and 10496 DF, p-value: < 2.2e-16

varImp(m\_full\_nhood\_final\_trainSet)

Overall

year 50.286080

beds 73.405694

baths\_na\_interpolation\_i 4.979902

sqft\_na\_interpolation\_i 19.515366

nhood\_cln\_alamo\_square 2.370560

nhood\_cln\_belmont\_san\_carlos 2.923067

nhood\_cln\_berkeley 4.757270

nhood\_cln\_berkeley\_north\_hills 4.192780

nhood\_cln\_bernal 5.994326

nhood\_cln\_brentwood\_oakley\_discovery\_bay 14.588623

nhood\_cln\_burlingame\_hillsborough 6.051314

nhood\_cln\_castro 10.167185

nhood\_cln\_CCSF 4.254436

nhood\_cln\_civic\_van\_ness 10.067523

nhood\_cln\_cole\_valley 7.963735

nhood\_cln\_concord\_pleasant\_hill\_martinez 6.890921

nhood\_cln\_cupertino 2.996279

nhood\_cln\_downtown 4.982439

nhood\_cln\_dublin\_pleasanton 4.390246

nhood\_cln\_emeryville 3.085723

nhood\_cln\_fairfield\_vacaville 10.606952

nhood\_cln\_financial\_district 15.691269

nhood\_cln\_foster\_city 3.381182

nhood\_cln\_gilroy 3.677193

nhood\_cln\_glen\_park 4.199739

nhood\_cln\_haight\_ashbury 5.002017

nhood\_cln\_hayes\_valley 8.481189

nhood\_cln\_hayward\_castro\_valley 6.350813

nhood\_cln\_healdsburg\_windsor 4.557858

nhood\_cln\_hercules\_pinole\_san\_pablo\_el\_sob 5.889315

nhood\_cln\_inner\_richmond 5.890313

nhood\_cln\_inner\_sunset 6.224615

nhood\_cln\_larkspur 4.977920

nhood\_cln\_los\_altos 9.212696

nhood\_cln\_los\_gatos 2.189914

nhood\_cln\_lower\_haight 4.851160

nhood\_cln\_lower\_pac\_hts 11.664164

nhood\_cln\_marina\_cow\_hollow 22.778447

nhood\_cln\_menlo\_park 6.171319

nhood\_cln\_mill\_valley 8.512455

nhood\_cln\_mission\_district 10.591445

nhood\_cln\_mountain\_view 5.061534

nhood\_cln\_napa\_county 5.770151

nhood\_cln\_nob\_hill 17.581295

nhood\_cln\_noe\_valley 12.503057

nhood\_cln\_NOPA 6.539373

nhood\_cln\_north\_beach\_telegraph\_hill 14.307942

nhood\_cln\_oakland\_east 8.326502

nhood\_cln\_oakland\_hills\_north\_montclair 3.573235

nhood\_cln\_oakland\_west 3.775952

nhood\_cln\_outer\_sunset 2.737181

nhood\_cln\_pacific\_heights 33.119278

nhood\_cln\_palo\_alto 12.690982

nhood\_cln\_parkside 3.314357

nhood\_cln\_petaluma 4.205159

nhood\_cln\_pittsburg\_antioch 16.355667

nhood\_cln\_portola 2.339311

nhood\_cln\_potrero\_hill 11.945104

nhood\_cln\_presidio\_hts\_laurel\_hts\_lake\_st 13.869118

nhood\_cln\_redwood\_city 3.761744

nhood\_cln\_redwood\_shores 2.622818

nhood\_cln\_richmond\_point\_annex 5.851330

nhood\_cln\_rohnert\_pk\_cotati 5.380219

nhood\_cln\_russian\_hill 25.280190

nhood\_cln\_russian\_river 2.461830

nhood\_cln\_san\_jose\_east 3.887840

nhood\_cln\_san\_jose\_south 4.113546

nhood\_cln\_san\_leandro 5.081609

nhood\_cln\_san\_mateo 5.182952

nhood\_cln\_santa\_rosa 8.923555

nhood\_cln\_saratoga 8.059973

nhood\_cln\_sausalito 9.081950

nhood\_cln\_sea\_cliff 5.205530

nhood\_cln\_SOMA\_south\_beach 37.359480

nhood\_cln\_tenderloin 2.278919

nhood\_cln\_tiburon\_belvedere 15.133048

nhood\_cln\_twin\_peaks 3.788925

nhood\_cln\_union\_city 5.060854

nhood\_cln\_USF\_anza\_vista 2.754241

nhood\_cln\_vallejo\_benicia 11.852649

nhood\_cln\_west\_portal\_forest\_hills 11.009902

nhood\_cln\_western\_addition 3.875419

# Note that the training set RMSE is 1021- comparable to the all-records model (RMSE=1031).

# Now run test cases through the training model to evaluate RMSE for these new cases:

testSet\_predictedValues <- predict(m\_full\_nhood\_final\_trainSet, newdata = testSet, type="response")

head(testSet\_predictedValues,25)

# Put the test dataset and predicted values together:

testSet <- cbind(testSet, testSet\_predictedValues)

str(testSet)

# Now get RMSE for test cases.

testSet$residuals <- testSet$price - testSet$testSet\_predictedValues

testSet$resid\_sq <- testSet$residuals^2

testSet\_resid\_sum\_sq <- sum(testSet$resid\_sq)

testSet\_resid\_sum\_sq

resid\_mean\_sq <- testSet\_resid\_sum\_sq/nrow(testSet)

resid\_mean\_sq

resid\_root\_mean\_sq <- sqrt(resid\_mean\_sq)

resid\_root\_mean\_sq

# The error of prediction for the test ($1,072) is a little higher than it was for

# the training set ($1,021). But the test error amount is within ~5% of the training

# model error, so we conclude that we have a reasonable model for price prediction.

# Use the model to predict listing price for a specific set of predictor values.

# To set up the data to predict a single case,

# take the first row of the final dataset and set all columns to zero.

# Then edit values and run through the model to get predicted values.

SFdata\_lm\_vars\_final\_for\_pred <- trainSet\_final[1,]

SFdata\_lm\_vars\_final\_for\_pred[SFdata\_lm\_vars\_final\_for\_pred != 0] <- 0

# check that all predictors are zero.

#SFdata\_lm\_vars\_final\_pred

# 1a. Apartment listing in 2022, one bed, one bath, 1000 sqft, in san\_jose\_east:

Apartment1 <- SFdata\_lm\_vars\_final\_for\_pred

Apartment1$year <- 2022

Apartment1$beds <- 1

Apartment1$baths\_na\_interpolation\_i <- 1

Apartment1$sqft\_na\_interpolation\_i <- 1000

Apartment1$nhood\_cln\_san\_jose\_east <- 1

# Check the predictor values

# Apartment1

# Predict the listing price.

predict(m\_full\_nhood\_final\_trainSet,Apartment1,type="response")

# 1b. What did this apartment go for 10 years ago?

Apartment1$year <- 2012

predict(m\_full\_nhood\_final\_trainSet,Apartment1,type="response")

# 1c. Same apartment in 2022, in pacific\_heights:

Apartment1$year <- 2022

Apartment1$nhood\_cln\_san\_jose\_east <- 0

Apartment1$nhood\_cln\_pacific\_heights <- 1

predict(m\_full\_nhood\_final\_trainSet,Apartment1,type="response")

1. Code for running the word clouds.

# Word cloud. For insight into apartment location and attributes.

# Whole-dataset word cloud.

str(SFdata)

# Prep the `title` column for text mining.

# First save into a new column- do not touch the original column.

# Here, clean as needed any characters not removed by other functions/commands:

remove\_mos <- c("Aug ", "Sep ", "Oct ", "Nov ", "Dec ", "map", "pic", "img")

SFdata$title\_wc <- str\_remove\_all(SFdata$title, paste(remove\_mos, collapse = "|"))

# Treat each submission/entry as a separate document.

meaning.corpus <- corpus(SFdata$title\_wc)

# Tokenize the strings so that each word is a token- `bag of words.`

# Remove punctuation, numbers, symbols.

meaning.tokens <- tokens(meaning.corpus, remove\_punct = TRUE, remove\_numbers = TRUE,

remove\_symbols = TRUE)

# Remove stopwords.

meaning.tokens <- tokens\_remove(meaning.tokens, stopwords("english"))

# get a document-term matrix.

meaning.dtm <- dfm(meaning.tokens)

# Get word cloud

textplot\_wordcloud(meaning.dtm, min\_count = 25)

# Get a function to create a word cloud for a specific neighborhood from the SFdata df.

# Make a new column (don`t touch the original column`), clean it & make the word cloud.

# Some nhoods have few rows of data, so high values of min\_wrd\_cnt may preclude a cloud.

# First, may be helpful to see all the nhoods:

# table(SFdata$nhood)

wrd\_cloud\_nhood <- function(nhood\_name,min\_wrd\_cnt) {

vect <- SFdata[SFdata$nhood == nhood\_name,]$title

remove\_mos <- c("Aug ", "Sep ", "Oct ", "Nov ", "Dec ","map","pic","img")

vect <- str\_remove\_all(vect, paste(remove\_mos, collapse = "|"))

# lowercase- combines words that differ only by case, and makes cleaning more feasible.

vect <- tolower(vect)

# Now convert to the needed objects

vect.corpus <- corpus(vect)

vect.tokens <- tokens(vect.corpus, remove\_punct = TRUE, remove\_numbers = TRUE,

remove\_symbols = TRUE)

vect.tokens <- tokens\_remove(vect.tokens, stopwords("english"))

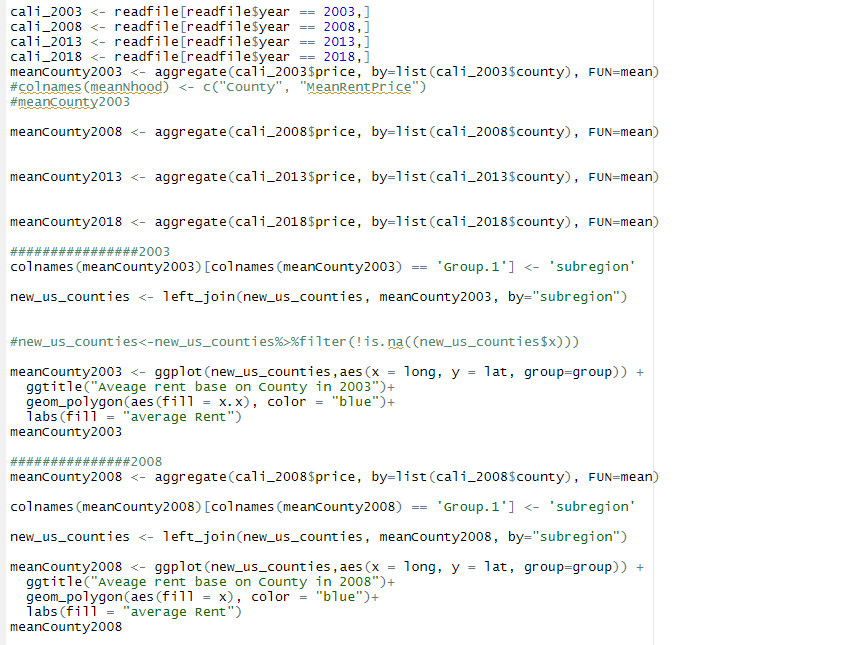
vect.dtm <- dfm(vect.tokens)

textplot\_wordcloud(vect.dtm, min\_count = min\_wrd\_cnt)

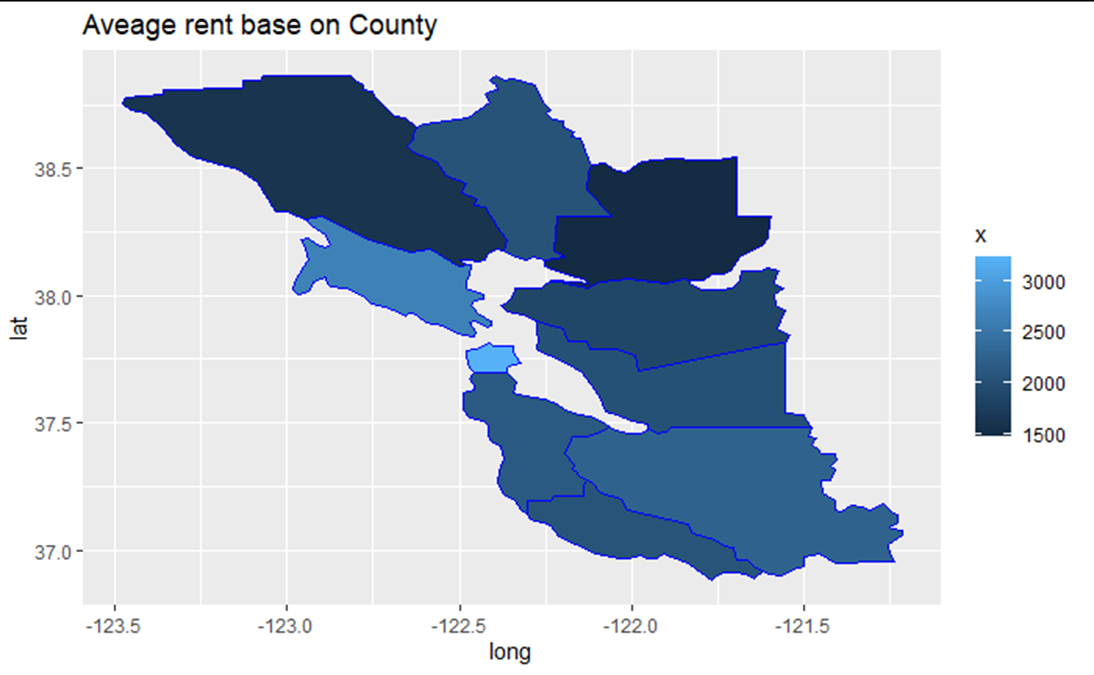
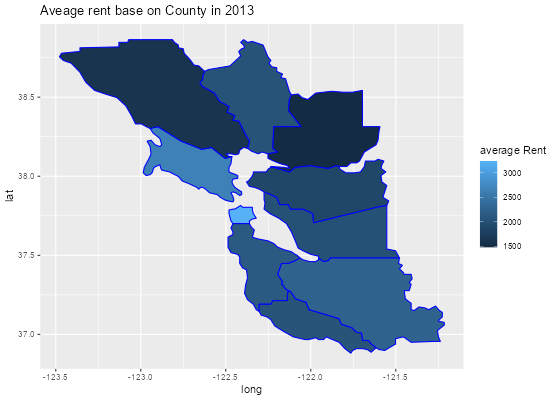
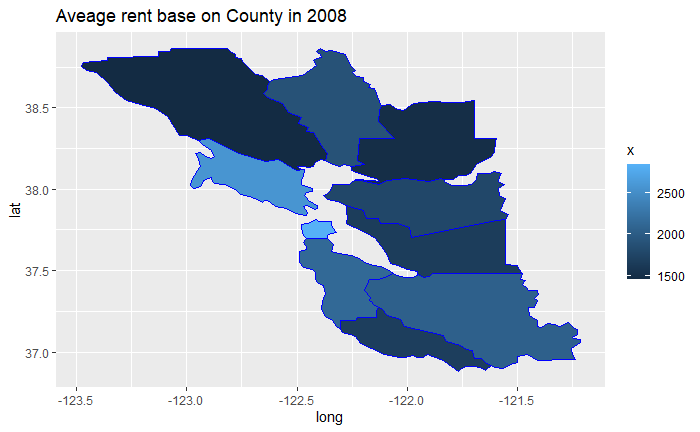
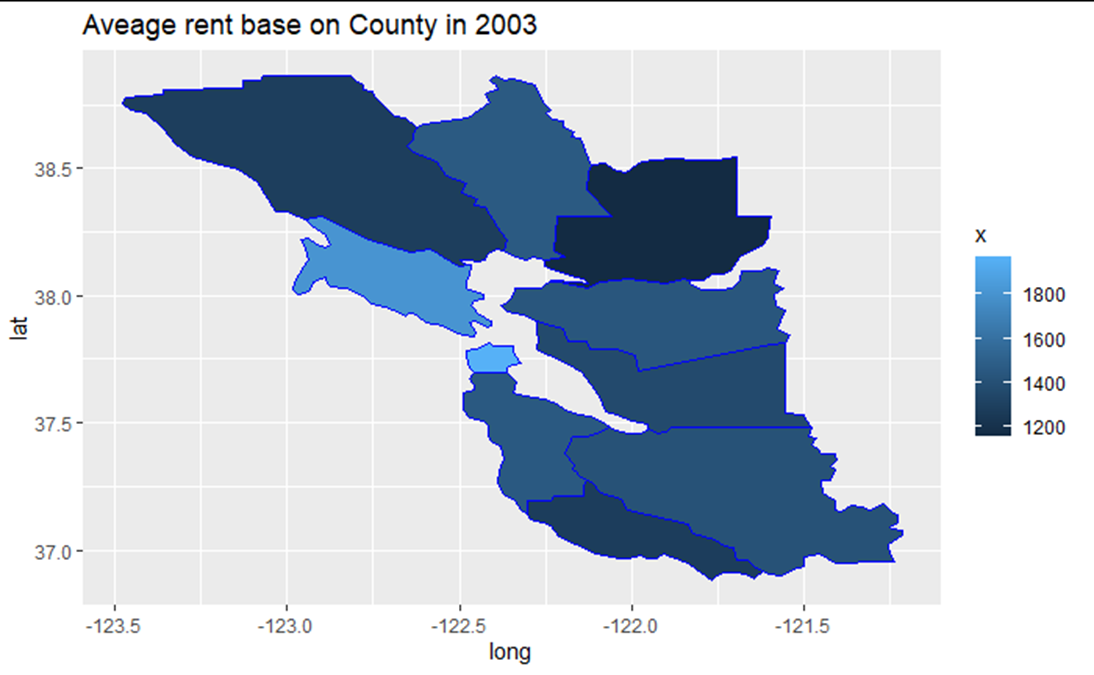
}

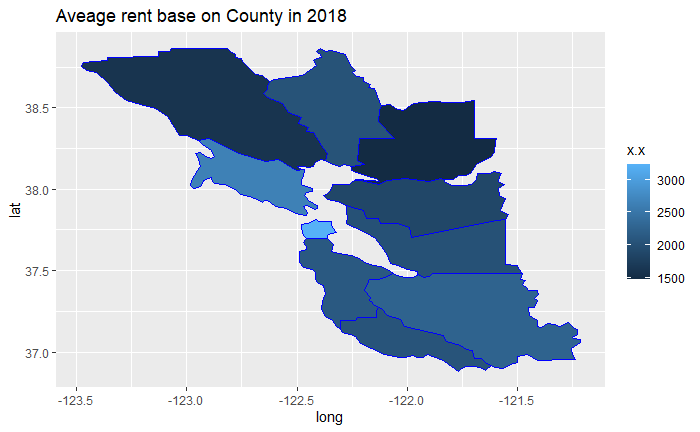
wrd\_cloud\_nhood("pacific heights",5)

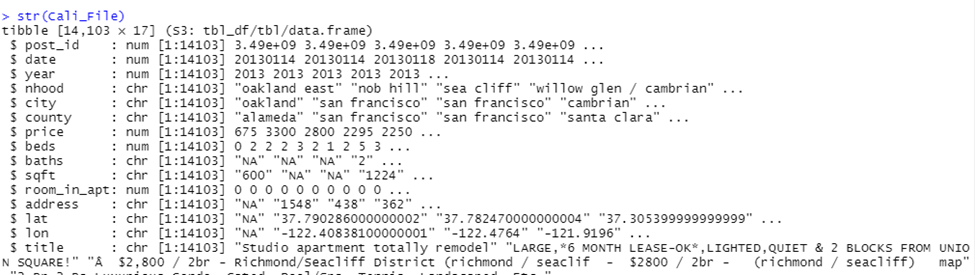
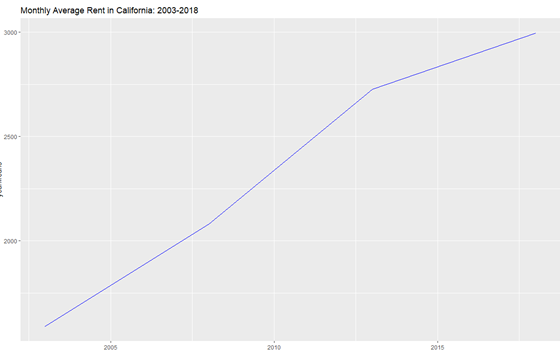
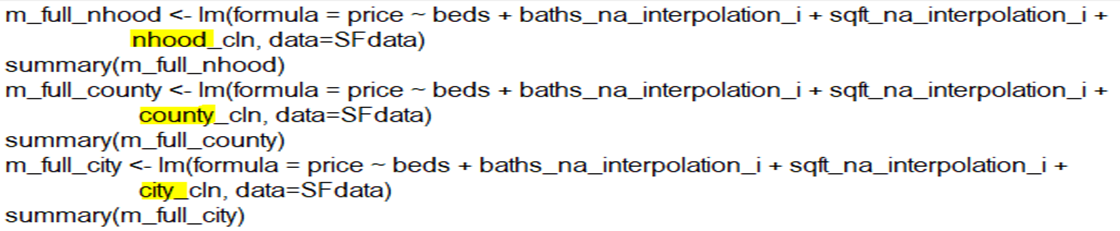
wrd\_cloud\_nhood("pittsburg / antioch",5)



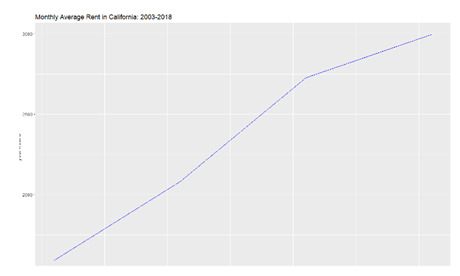
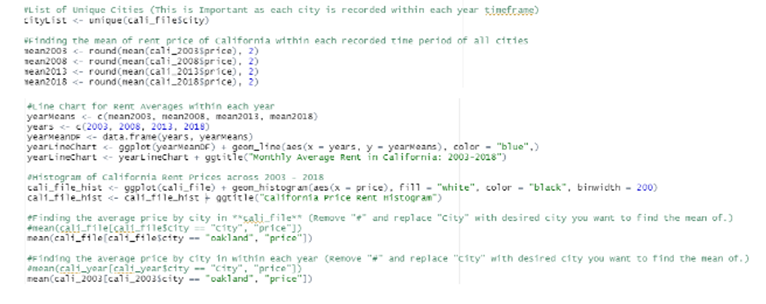
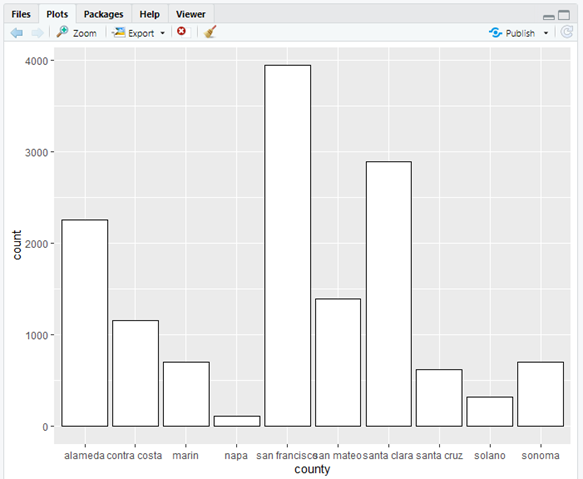


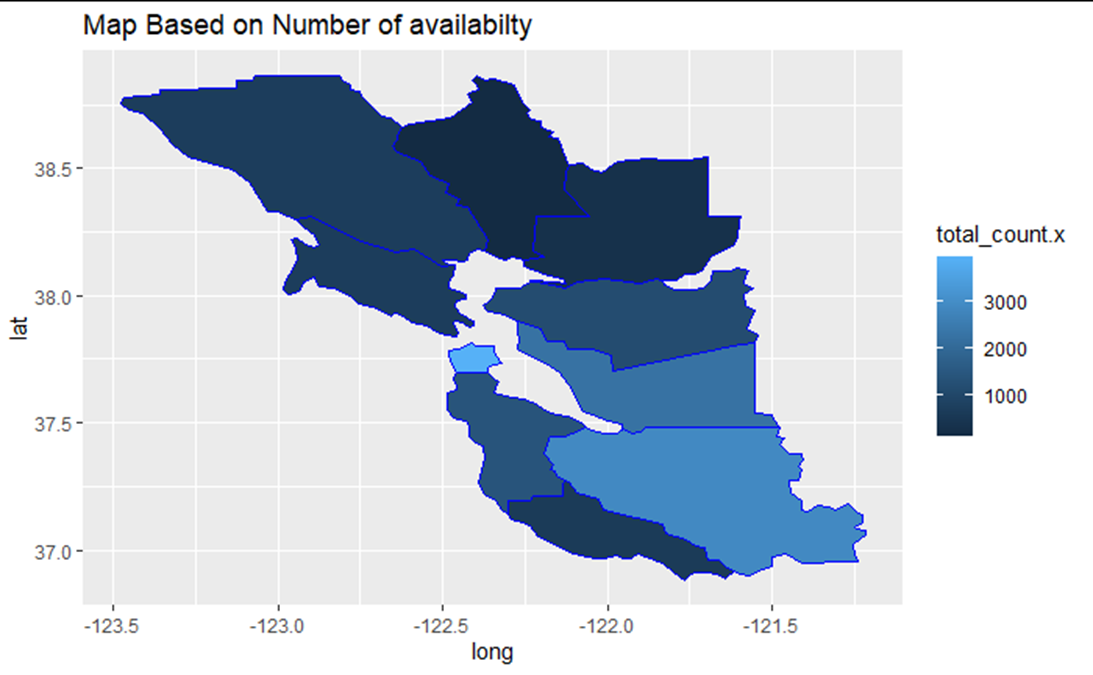
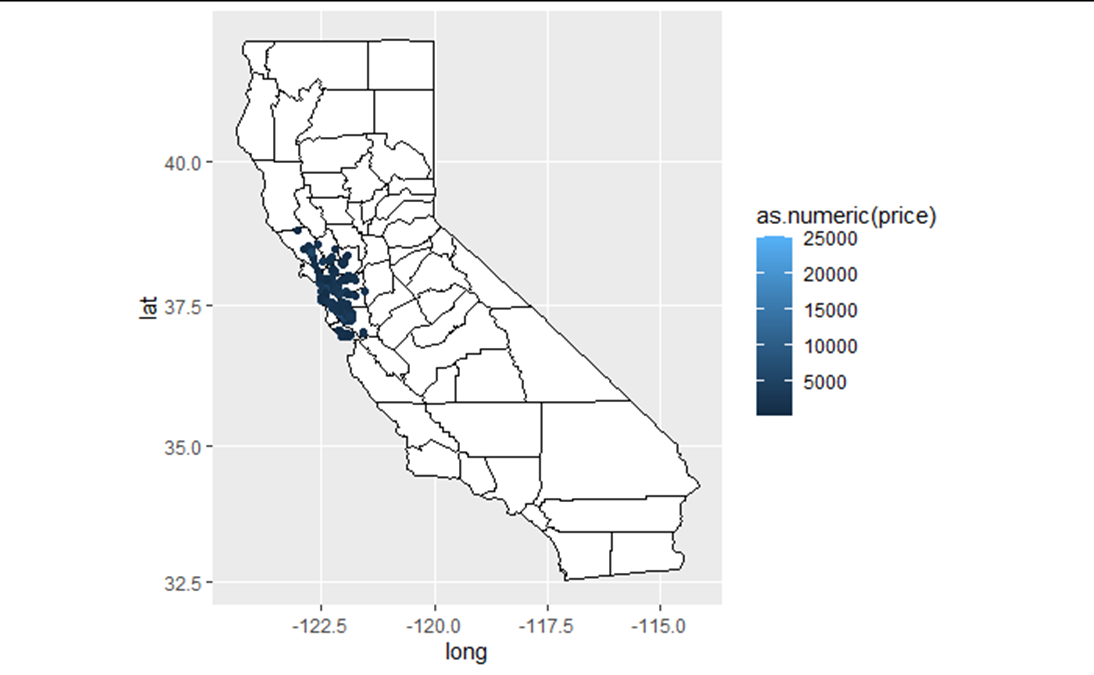
** **

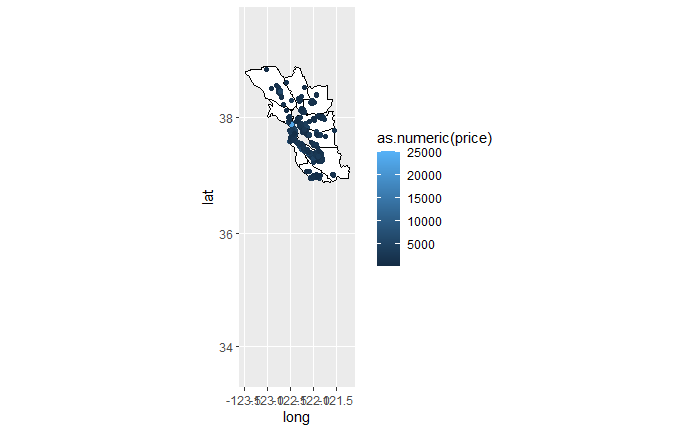
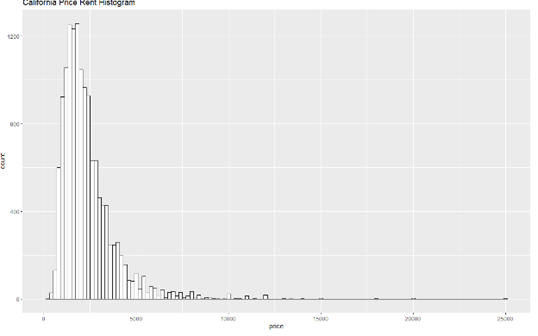
****

****

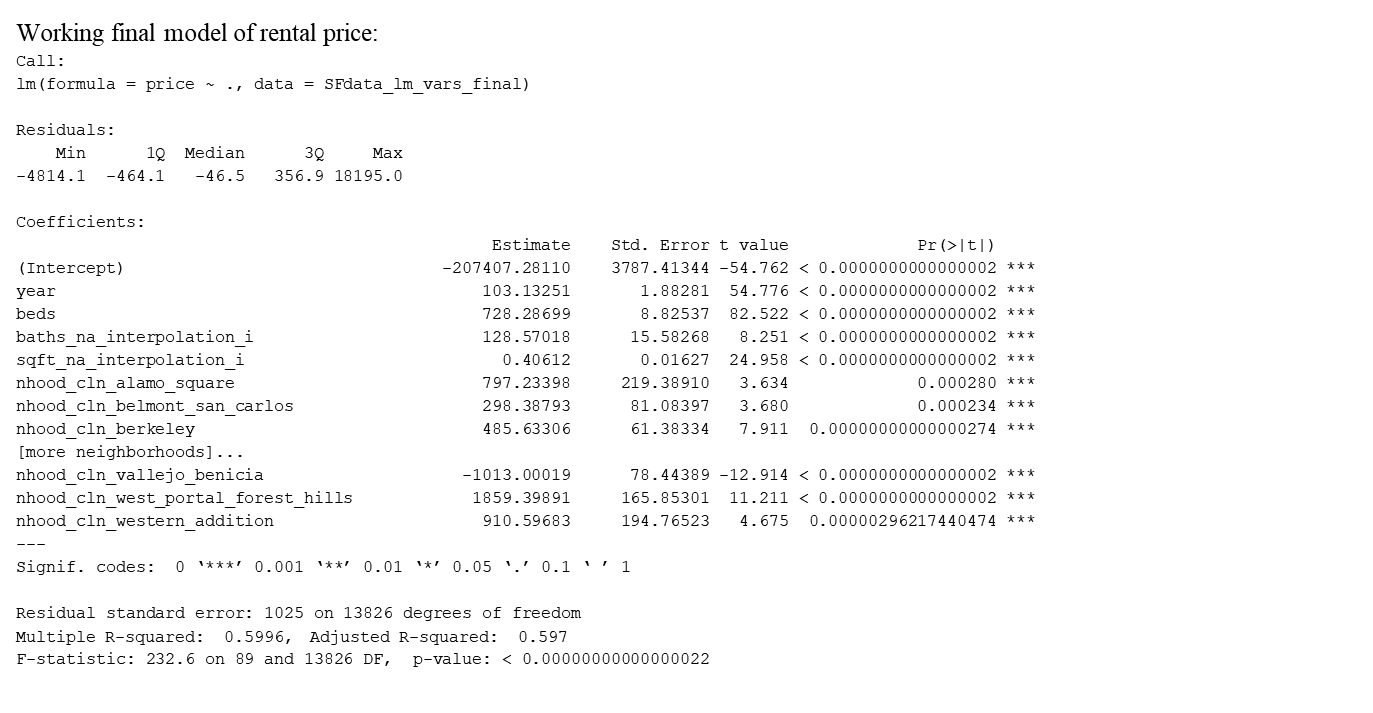
## 

** **

** **

****

****

****

Descriptive Statistics Code:

#Important Rent Price Averages

#Year Average Rent Price

meanYear <- aggregate(cali\_file$price, by=list(cali\_file$year), FUN=mean)

colnames(meanYear) <- c("Year", "MeanRentPrice")

meanYear$`MeanRentPrice` <- round(meanYear$`MeanRentPrice`, digits = 2)

#Neighborhood Average Rent Price

meanNhood <- aggregate(cali\_file$price, by=list(cali\_file$nhood), FUN=mean)

colnames(meanNhood) <- c("Neighborhood", "MeanRentPrice")

meanNhood$MeanRentPrice <- round(meanNhood$MeanRentPrice, digits = 2)

#County Average Rent Price

meanCounty <- aggregate(cali\_file$price, by=list(cali\_file$county), FUN=mean)

colnames(meanCounty) <- c("County", "MeanRentPrice")

meanCounty$MeanRentPrice <- round(meanCounty$MeanRentPrice, digits = 2)

#City Average Rent Price

meanCities <- aggregate(cali\_file$price, by=list(cali\_file$city), FUN=mean)

colnames(meanCities) <- c("City", "MeanRentPrice")

meanCities$MeanRentPrice <- round(meanCities$MeanRentPrice, digits = 2)

#Population Stats

cityStats <- cbind(cityStats, meanCities$MeanRentPrice)

colnames(cityStats)[10] <- "MeanRentPrice"

#-----------------------------#

#Average Rent Price by City & Year

meanCities2003 <- aggregate(cali\_2003$price, by=list(cali\_2003$city), FUN=mean)

colnames(meanCities2003) <- c("City", "MeanRentPrice")

meanCities2003 <- data.frame(meanCities2003, 2003)

colnames(meanCities2003)[3] <- "Year"

meanCities2003$MeanRentPrice <- round(meanCities2003$MeanRentPrice, digits = 2)

meanCities2008 <- aggregate(cali\_2008$price, by=list(cali\_2008$city), FUN=mean)

colnames(meanCities2008) <- c("City", "MeanRentPrice")

meanCities2008 <- data.frame(meanCities2008, 2008)

colnames(meanCities2008)[3] <- "Year"

meanCities2008$MeanRentPrice <- round(meanCities2008$MeanRentPrice, digits = 2)

meanCities2013 <- aggregate(cali\_2013$price, by=list(cali\_2013$city), FUN=mean)

colnames(meanCities2013) <- c("City", "MeanRentPrice")

meanCities2013 <- data.frame(meanCities2013, 2013)

colnames(meanCities2013)[3] <- "Year"

meanCities2013$MeanRentPrice <- round(meanCities2013$MeanRentPrice, digits = 2)

meanCities2018 <- aggregate(cali\_2018$price, by=list(cali\_2018$city), FUN=mean)

colnames(meanCities2018) <- c("City", "MeanRentPrice")

meanCities2018 <- data.frame(meanCities2018, 2018)

colnames(meanCities2018)[3] <- "Year"

meanCities2018$MeanRentPrice <- round(meanCities2018$MeanRentPrice, digits = 2)

#-----------------------------#

#Average Rent Price By County & Year

meanCounty2003 <- aggregate(cali\_2003$price, by=list(cali\_2003$county), FUN=mean)

colnames(meanCounty2003) <- c("County", "MeanRentPrice")

meanCounty2003 <- data.frame(meanCounty2003, 2003)

colnames(meanCounty2003)[3] <- "Year"

meanCounty2003$MeanRentPrice <- round(meanCounty2003$MeanRentPrice, digits = 2)

meanCounty2008 <- aggregate(cali\_2008$price, by=list(cali\_2008$county), FUN=mean)

colnames(meanCounty2008) <- c("County", "MeanRentPrice")

meanCounty2008 <- data.frame(meanCounty2008, 2008)

colnames(meanCounty2008)[3] <- "Year"

meanCounty2008$MeanRentPrice <- round(meanCounty2008$MeanRentPrice, digits = 2)

meanCounty2013 <- aggregate(cali\_2013$price, by=list(cali\_2013$county), FUN=mean)

colnames(meanCounty2013) <- c("County", "MeanRentPrice")

meanCounty2013 <- data.frame(meanCounty2013, 2013)

colnames(meanCounty2013)[3] <- "Year"

meanCounty2013$MeanRentPrice <- round(meanCounty2013$MeanRentPrice, digits = 2)

meanCounty2018 <- aggregate(cali\_2018$price, by=list(cali\_2018$county), FUN=mean)

colnames(meanCounty2018) <- c("County", "MeanRentPrice")

meanCounty2018 <- data.frame(meanCounty2018, 2018)

colnames(meanCounty2018)[3] <- "Year"

meanCounty2018$MeanRentPrice <- round(meanCounty2018$MeanRentPrice, digits = 2)

#-----------------------------#

#Average Rent Price By Neighborhood & Year

meanNhood2003 <- aggregate(cali\_2003$price, by=list(cali\_2003$nhood), FUN=mean)

colnames(meanNhood2003) <- c("Neighborhood", "MeanRentPrice")

meanNhood2003 <- data.frame(meanNhood2003, 2003)

colnames(meanNhood2003)[3] <- "Year"

meanNhood2003$MeanRentPrice <- round(meanNhood2003$MeanRentPrice, digits = 2)

meanNhood2008 <- aggregate(cali\_2008$price, by=list(cali\_2008$nhood), FUN=mean)

colnames(meanNhood2008) <- c("Neighborhood", "MeanRentPrice")

meanNhood2008 <- data.frame(meanNhood2008, 2008)

colnames(meanNhood2008)[3] <- "Year"

meanNhood2008$MeanRentPrice <- round(meanNhood2008$MeanRentPrice, digits = 2)

meanNhood2013 <- aggregate(cali\_2013$price, by=list(cali\_2013$nhood), FUN=mean)

colnames(meanNhood2013) <- c("Neighborhood", "MeanRentPrice")

meanNhood2013 <- data.frame(meanNhood2013, 2013)

colnames(meanNhood2013)[3] <- "Year"

meanNhood2013$MeanRentPrice <- round(meanNhood2013$MeanRentPrice, digits = 2)

meanNhood2018 <- aggregate(cali\_2018$price, by=list(cali\_2018$nhood), FUN=mean)

colnames(meanNhood2018) <- c("Neighborhood", "MeanRentPrice")

meanNhood2018 <- data.frame(meanNhood2018, 2018)

colnames(meanNhood2018)[3] <- "Year"

meanNhood2018$MeanRentPrice <- round(meanNhood2018$MeanRentPrice, digits = 2)

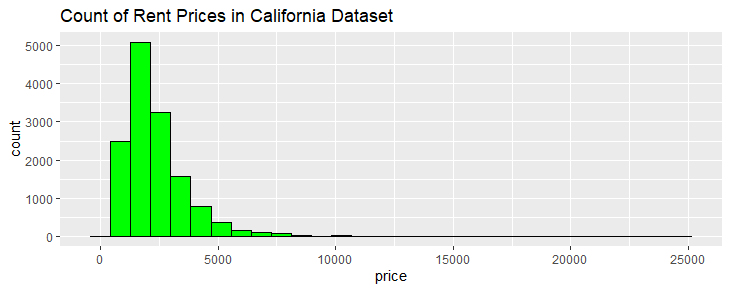
#-----------------------------#

#Combined Dataframes (Location type by year)

meanCitiesYear <- rbind(meanCities2003, meanCities2008, meanCities2013, meanCities2018)

meanCountyYear <- rbind(meanCounty2003, meanCounty2008, meanCounty2013, meanCounty2018)

meanNhoodYear <- rbind(meanNhood2003, meanNhood2008, meanNhood2013, meanNhood2018)



# 7. References

Badger, E. (2016, September 1). *What more than 1 million Craigslist rental listings tell us about the housing market*. The Washington Post. Retrieved from <https://www.washingtonpost.com/news/wonk/wp/2016/09/01/what-more-than-1-million-craigslist-rental-listings-tell-us-about-the-housing-market/>

Boeing, G., & Waddell, P. (2017). New insights into rental housing markets across the United States: Web scraping and analyzing craigslist rental listings. *Journal of Planning Education and Research, 37*(4), 457-476. https://doi.org/10.1177/0739456X16664789

Boeing, G., Wegmann, J., & Jiao, J. (2020). Rental housing spot markets: How online information exchanges can supplement transacted-rents data. *Journal of Planning Education and Research*, published online ahead of print. https://doi.org/10.1177/0739456X20904435

Nelson, A. C. (2016). On the plausibility of a 53-percent homeownership rate by 2050. *Cityscape, 18*(1), 125–130. https://www.jstor.org/stable/26328244

Pennington, K. (2018). *Bay Area Craigslist Rental Housing Posts, 2000-2018*. Retrieved from https://github.com/katepennington/historic\_bay\_area\_craigslist\_housing\_posts/blob/master/clean\_2000\_2018.csv.zip

Saltz, J. S. & Stanton, J. M. (2022). *Data science for business with R*. Thousand Oaks, CA: Sage Publications.

Williams, P. (2022, December 12). *How did the housing market get so unaffordable for so many?* The Wall Street Journal. Retrieved from https://www.wsj.com/articles/housing-market-prices-unaffordable-11670604049