Final\_IS457\_122

Class ID 122

2019-04-15

Whole Tale URL: <https://dashboard.wholetale.org/run/5cbfee112744a50001c60b24>

I wasn’t sure how to publish my tale, but here is the URL.

## Part 1

#### Q1:

# Loading the dataset  
Airbnb\_Sydney <- read.csv(file = "Airbnb Sydney.csv")

# Exploring dataset a little bit  
head(Airbnb\_Sydney)

## id  
## 1 12351  
## 2 14250  
## 3 15253  
## 4 20865  
## 5 38073  
## 6 39348  
## description  
## 1 Come stay with Vinh & Stuart (Awarded as one of Australia's top hosts by Airbnb CEO Brian Chesky & key shareholder Ashton Kutcher. We're Sydney's #1 reviewed hosts too). Find out why we've been positively reviewed 500+ times. Message us and talk first BEFORE you make any reservation request - And please read our listing to the end (hint hint). Everything you need to know is there. We're pretty relaxed hosts, and we fully appreciate staying with someone else, in their home home, is not for every-one. This is not a business, or a hotel. We're casual Airbnb hosts, not hoteliers. If you're just looking for an alternative to an expensive hotel, then we're not for you. Here you'll be treated in the same way we treat family & friends when they stay. So... no fluffy bathrobes... Please say hello and message us \*BEFORE\* you make your reservation request... It'll help speed things up, and smooth things out... Please read our listing all the way to the end. It will make getting a confirmed reserv  
## 2 Beautifully renovated, spacious and quiet, our 3 Bedroom, 3 Bathroom home is only a 10 minute walk to beaches in Fairlight or Forty Baskets, or a 30 minute walk to Manly via the coastal promenade, or an Express bus runs every 20 mins at your door. Our home is a thirty minute walk along the seashore promenade to Manly, one of Sydney's most beautiful beaches, with its village restaurants, cafes, and shopping. If you prefer more variety, the Manly ferry will take you to the Sydney CBD in 15 minutes. The residence is sited in a sought-after family-friendly street only a short stroll to nearby North Harbour reserve and Forty Baskets cafe and beach. It's a short walk further to express CBD buses, ferries, and Manly entertainment. Or there is a bus (#131 or #132) around the corner that drops you in Manly in 8 minutes. Our home features a stainless steel galley kitchen, including Ilve oven and gas cooktop. We have two separate living areas on the ground floor. The front lounge enjoys P&O  
## 3 Penthouse living in a great central location: You will be staying in a unique apartment on the top floor of a centrally located boutique building. A spacious apartment spread over 2 levels, the space offers my guests a high level of privacy and security. The room has its own bathroom and balcony and the whole apartment is furnished to a luxury standard in a contemporary style with all mod cons. Location is one if the best in the city with easy walking to everything you may need. A charming two-level, two-bedroom, two-bathroom duplex apartment on the border of East Sydney, Darlinghurst and Kings Cross with everything that Sydney has to offer within easy walking distance : Sydney CBD 10mins, Hyde Park 5 mins, Oxford St 10 mins, Kings Cross 5mins The apartment is spacious, elegantly decorated in a modern contemporary style and very well equipped with all mod cons. The room is situated on the lower floor, it is a spacious bedroom with large built in robes, own fridge and tea and coffe  
## 4 Hi! We are a married professional couple with 2 kids. When making a booking, please tell us a little bit about yourselves (ages & professions of everyone in your group), the purpose of your visit and finally, your check in & out times. Thanks! HOUSE : \_\_\_\_\_\_\_\_\_ \* DUCTED AIR CONDITIONING IN ALL ROOMS \* BEDROOM 1 : QUEEN BED \* BEDROOM 2 : QUEEN BED \* BEDROOM 3 : QUEEN BED \* BATHROOM : SHOWER, BATH & TOILET \* KITCHEN : OVEN, STOVE TOP, DISHWASHER, MICROWAVE, FRIDGE \* WASHING MACHINE & DRYER STUDIO : \_\_\_\_\_\_\_\_\_\_ \* BEDROOM : QUEEN BED \* BATHROOM : SHOWER & TOILET \* AIR CONDITIONER (in Studio area where there is the Queen bed) \* KITCHENETTE / STUDY : Oven, dishwasher, bar fridge & washing machine, sofabed (wide single), computer, TV & desk. \* There is internal access to the studio from the house but it can also be closed off to the front of the property with its own side access. (Website hidden by Airbnb) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* LOCATION : \* 5 minute walk   
## 5 Welcome to my sanctuary - a bright, comfortable one bedroom apartment in North Sydney. Free Wifi, heated pool/jacuzzi and everything else that you will need to make your stay in Sydney very comfortable. Enjoy this fabulous Home away from home, and have a fantastic stay in Sydney! The apartment is within walking distance of restaurants and shops, Luna Park and the North Sydney business district. Access to the Sydney CBD is easy by bus, train, taxi or ferry. It is also a short bus ride to the famous Balmoral Beach or Taronga Zoo. My apartment is situated in North Sydney which is 3 kms from the Sydney CBD. Here are some details about the apartment: You'll enjoy being centrally located just a couple blocks away from the train station so you can go anywhere quickly in Sydney. The apartment also features several windows that let in tons of natural light. It is comfortable and fully stocked. Here's what I have here: LIVING ROOM: 50" LCD TV DVD / blu ray player CD/Radio/Blue tooth syncing w  
## 6 Fully self-contained sunny studio apartment. 10mn to walk to Bondi beach. Bus to city at the door. Private 13m swimming pool. Sunny, studio apartment . Private terrace. bus at door to Bondi Junction and City Ground floor 1 bedroom with double bed plus kitchenette & study desk. own shower & toilet, share laundry, kitchen facilities Swimming pool 13m. Separate security private entrance Private entrance. Ground floor. Happy to indicate you the best spots for walking, dining, entertaining and best sightseeing location in Sydney. Upmarket area. Very nice and quiet neighbourhood . Very safe place. Bus at the door for the city.  
## neighborhood\_overview  
## 1 Pyrmont is an inner-city village of Sydney, only about 2kms from the Sydney CBD (Central Business District) / Core, right next door to Darling Harbour and Chinatown. https://www.airbnb.com.au/locations/sydney/pyrmont Pyrmont has a relaxed community feel with an inner city vibe. Pyrmont is only about 2kms (10 - 15 mins walk) from the centre of Sydney with an extensive range of local restaurants, wine bars, and pubs. There's some seriously good baristas and cafes right close to our home (Go and say hi to Damien & Tim at Bar Zini - it's one of our local faves). There's five star fine dining right through to greasy-spoon takeaways with some of Australia's finest dining restaurants within easy reach. \* Darling Harbour and Cockle Bay \* Exhibition and Convention Centres \* Sydney Fish Markets \* Pyrama Point Park \* Jones Bay Wharf \* The Star (Casino) with Food Halls, Bars, and the Lyric Theatre \* Powerhouse Museum \* National Maritime Museum Also (back on the food - notice a theme?): \* Two groce  
## 2 Balgowlah Heights is one of the most prestigious areas on the Northern Beaches. Filled with seaside character and a boutique way of living, this suburb offers everything you could need. Located approximately 11km from the CBD, and only 2 kms from Manly, Balgowlah is surrounded by pristine water frontage including North Harbour, Forty Baskets, and Sydney Harbour. Filled with a vast array of public parks, pools, marinas, beaches, sporting facilities and Sydney Harbour National Park. Local Amenities •Forty Baskets Beach, Reef Beach, North Harbour Park, 40 Beans Cafe, Clontarf Beach, Castle Rock. •Nearby Spit to Manly walking track that leads left along the promenade to Manly or right through scenic Sydney Harbour National Park around to Clontarf. •Balgowlah Heights shops, offering a supermarket, delicatessen, boutique shops, cafes and more •Nearby Stockland Mall offers a vast range of cafes, supermarkets, eateries, boutique fashion stores, home wares shops and Fitness First Gym •Array  
## 3 The location is really central and there is number of things to do and see all within a few kilometres; Stanley St (Sydney’s little Italy) is just around the corner which has some great restaurants and a real European feel. Darlinghurst is wall to wall of cafes, bars and restaurants to suite all tastes and budgets. Woolloomooloo marina is at the bottom of the hill with its ritzy restaurants and famous residents (Russell Crow has the penthouse at end of wharf) it is beautiful to hang out on a nice evening. The Australian Museum and The Art Gallery of NSW have both interesting exhibitions and evening events, The Bridge can be seen from my corner and the Opera House is a pleasant stroll through the Botanical Gardens and Domain. Chinatown and Darling Harbour are just at the back of the CBD and Sydney colourful nightlife of both Kings Cross & Oxford St are on your doorstep. I also have lots of more information on things to do while in Sydney just ask me. On booking you will receive my hou  
## 4 BALMAIN is an older inner city village / suburb with numerous cafes, restaurants, parks, walks around the harbour, older-style pubs, markets, etc. Our house situated between Balmain and Rozelle shopping centres, in a quiet street with a small park at the end of it that has gated play equipment for small children.  
## 5 North Sydney, on Sydney's lower North Shore, starts at north end of the Harbour Bridge and is around 3km from the CBD and also closely located on route to Sydney's famous Manly beach and other North shore village suburbs including Neutral Bay, Crows Nest, McMahons Point, Kirribilli Chatswood and Mosman (Balmoral Beach & Taronga Zoo). North Sydney is the second largest business centre to Sydney's CBD and located only a 5 minute drive directly over the Harbour Bridge, making is an excellent alternative to the hustle and bustle of the CBD - whilst still being centrally located. The area is dominated by the IT and advertising industries and café scene during the week and benefits from the quieter peaceful surroundings at the weekend as the corporates go home to the suburbs. North Sydney is a prosperous area with spectacular waterfront real estate and is location of official residences of the Australian Prime Minister and Governor-General at Kirribilli Point. The area is easily accessible b  
## 6 Upmarket area. Very nice and quiet neighbourhood . Very safe place.  
## house\_rules  
## 1 We look forward to welcoming you to stay you just as we would our family and friends. "Farm Gate Rules" - if a door is open, leave it open, if a door is closed, please leave it closed. We'd ask that you'd please not eat in your room, or smoke inside the house. We've a kitchen and dining room for meals (and of course, feel free to use the fridge) and a sheltered, undercover area outside that you're more than welcome to use if you smoke. We tend to work from home, meet clients in our home office, and work by phone too. So this means that our home would be better suited to guests wanting to be out sightseeing during the day, rather than spending the days inside. (But why would you want to stay inside all day when there's so much to see and do anyway?) Every experience we've ever had with Airbnb has been a positive one. Whether we've been hosting, or staying as guests, we've met kind & considerate people, had interesting conversations, and made great new acquaintances & friends. We don't   
## 2 Standard Terms and Conditions of Temporary Holiday Accommodation Note: Variances can be agreed on but only by arrangement with the owner in writing. Payment of booking constitutes the clients acceptance of these Terms and Conditions. Balance of the rental amount must be received in full according to AIRBNB policies. If not the owner has the right to cancel the booking and attempt to re let it. The owners will make every effort to ensure the property is available as booked. However the owners reserve the right to make alterations to bookings due to unforeseen circumstances. To maintain a good standard for our guests we require certain conditions to be complied with. We appreciate most will respect our property but the occasional abuse requires that we state the following conditions. Number of Guests should not exceed 6 adults or subsequently agreed in writing or email, and no more than 8 people in the house at one time. Fees will apply for excess guests not agreed with the owners in adv  
## 3 I am fairly easygoing and will try to accomodate guests reasonable requests. I ask that guest treat my home with respect. No Smoking inside of Apartment No additional overnight guests.  
## 4 PLEASE ENJOY YOURSELVES WITHOUT MAKING TOO MUCH NOISE AS WE HAVE VERY GOOD NEIGHBOURS. NO SMOKING INSIDE THE HOUSE. SMOKING ALLOWED IN OUTSIDE AREAS ONLY. PLEASE LET US KNOW WHEN BOOKING, IF YOU PLAN TO BRING YOUR PET. PLEASE CLEAN THE BBQ AFTER USE OR AN ADDITIONAL FEE MAY BE INCURRED. THANKS!  
## 5 House Rules: •Smoking permitted outside only with the sliding doors closed. If smoking is detected in the apartment an additional cleaning fee will apply of $350. •When using the BBQ, please do so with the balcony doors closed. •On arrival you will be given two sets of keys. Each set of keys contain a security fob which allows you entry into the building and to my level. If these keys are lost, the replacement cost is $200 for each set. •Please remove shoes whilst inside the apartment as it's fully carpeted. •This is a residential building, so no parties are allowed. •For your own safety, please do not sleep with the gas heater running. •Kitchen knives and wooden chopping boards are not to be placed in the dishwasher. •Please switch off all lights and gas heater when you are not in the apartment or before leaving. \*\*\*ALL BREAKAGES AND ANY DAMAGE MUST BE PAID FOR & PLEASE DO NOT MOVE ANY FURNITURE\*\*\* On Exit: Empty fridge and take all your garbage out Turn on dishwasher Place all  
## 6 Only quiet people. No parties aloud.  
## host\_id host\_since host\_response\_time host\_response\_rate  
## 1 17061 5/14/09 within a few hours 100%  
## 2 55948 11/20/09 within a few hours 90%  
## 3 59850 12/3/09 within an hour 100%  
## 4 64282 12/19/09 within a day 100%  
## 5 103476 4/4/10 N/A N/A  
## 6 168828 7/17/10 N/A N/A  
## host\_is\_superhost  
## 1 f  
## 2 f  
## 3 f  
## 4 t  
## 5 f  
## 6 f  
## host\_verifications  
## 1 ['email', 'phone', 'manual\_online', 'reviews', 'manual\_offline', 'offline\_government\_id', 'government\_id', 'work\_email']  
## 2 ['email', 'phone', 'reviews', 'jumio', 'offline\_government\_id', 'government\_id']  
## 3 ['email', 'phone', 'facebook', 'reviews', 'jumio', 'offline\_government\_id', 'government\_id']  
## 4 ['email', 'phone', 'reviews', 'jumio', 'government\_id', 'work\_email']  
## 5 ['email', 'phone', 'facebook', 'reviews', 'jumio', 'government\_id']  
## 6 ['email', 'phone', 'facebook', 'reviews', 'jumio', 'offline\_government\_id', 'selfie', 'government\_id', 'identity\_manual']  
## host\_identity\_verified city zipcode property\_type  
## 1 t Pyrmont 2009 Townhouse  
## 2 t Balgowlah 2093 House  
## 3 t Darlinghurst 2010 Apartment  
## 4 t Balmain 2041 House  
## 5 t North Sydney 2060 Apartment  
## 6 f North Bondi 2026 Guest suite  
## room\_type accommodates bathrooms bedrooms beds bed\_type  
## 1 Private room 2 1 1 1 Real Bed  
## 2 Entire home/apt 6 3 3 3 Real Bed  
## 3 Private room 2 1 1 1 Real Bed  
## 4 Entire home/apt 8 2 4 4 Real Bed  
## 5 Entire home/apt 2 1 0 1 Real Bed  
## 6 Entire home/apt 2 1 1 1 Real Bed  
## amenities  
## 1 {TV,Internet,Wifi,"Air conditioning","Paid parking off premises",Breakfast,Heating,"Smoke detector","Carbon monoxide detector","First aid kit","Safety card","Fire extinguisher",Essentials,Shampoo,"Lock on bedroom door","24-hour check-in",Hangers,"Hair dryer",Iron,"Laptop friendly workspace","translation missing: en.hosting\_amenity\_49","translation missing: en.hosting\_amenity\_50","Private entrance","Hot water","Patio or balcony","Garden or backyard","Luggage dropoff allowed","Well-lit path to entrance","Host greets you"}  
## 2 {TV,Wifi,"Air conditioning",Kitchen,"Pets live on this property",Cat(s),"Free street parking",Heating,Washer,Dryer,"Smoke detector",Essentials,Shampoo,Hangers,"Hair dryer",Iron,"Laptop friendly workspace","Hot water","Luggage dropoff allowed",Other}  
## 3 {TV,"Cable TV",Internet,Wifi,"Air conditioning",Kitchen,"Paid parking off premises","Pets allowed","Pets live on this property",Dog(s),"Free street parking","Buzzer/wireless intercom",Heating,Washer,Dryer,"Smoke detector","First aid kit","Fire extinguisher",Essentials,Shampoo,"24-hour check-in",Hangers,"Hair dryer",Iron,"Laptop friendly workspace","translation missing: en.hosting\_amenity\_49","translation missing: en.hosting\_amenity\_50","Self check-in",Lockbox,"Hot water","Bed linens","Extra pillows and blankets",Microwave,"Coffee maker",Refrigerator,Dishwasher,"Dishes and silverware","Cooking basics",Oven,Stove,"Patio or balcony","Luggage dropoff allowed","Well-lit path to entrance"}  
## 4 {TV,Internet,Wifi,"Air conditioning",Kitchen,"Pets allowed","Pets live on this property",Cat(s),"Indoor fireplace",Heating,"Family/kid friendly",Washer,Dryer,"Smoke detector","First aid kit",Essentials,Shampoo,"24-hour check-in",Hangers,"Hair dryer",Iron,"Laptop friendly workspace","Private entrance"}  
## 5 {TV,"Cable TV",Wifi,"Air conditioning",Pool,Kitchen,"Free parking on premises",Breakfast,Elevator,"Hot tub","Buzzer/wireless intercom",Heating,"Family/kid friendly",Washer,"Smoke detector","First aid kit",Essentials,Shampoo,"24-hour check-in",Hangers,"Hair dryer",Iron,"translation missing: en.hosting\_amenity\_50"}  
## 6 {Internet,Wifi,Pool,Kitchen,"Free street parking","Buzzer/wireless intercom",Heating,"Smoke detector",Essentials,Hangers,Iron,"Hot water",Microwave,"Coffee maker",Refrigerator,"Dishes and silverware","Cooking basics","BBQ grill","Garden or backyard","Long term stays allowed","Host greets you"}  
## price cleaning\_fee guests\_included extra\_people minimum\_nights  
## 1 $100.00 $55.00 2 $395.00 2  
## 2 $471.00 $100.00 6 $40.00 5  
## 3 $109.00 1 $10.00 2  
## 4 $450.00 6 $0.00 7  
## 5 $159.00 $250.00 2 $25.00 2  
## 6 $84.00 $90.00 1 $10.00 5  
## number\_of\_reviews review\_scores\_rating review\_scores\_accuracy  
## 1 493 95 10  
## 2 1 100 10  
## 3 300 88 9  
## 4 15 96 9  
## 5 63 97 10  
## 6 6 87 8  
## review\_scores\_cleanliness review\_scores\_checkin  
## 1 9 10  
## 2 10 10  
## 3 9 9  
## 4 9 9  
## 5 10 10  
## 6 8 9  
## review\_scores\_communication review\_scores\_location review\_scores\_value  
## 1 10 10 10  
## 2 8 10 10  
## 3 9 9 9  
## 4 10 10 9  
## 5 10 9 9  
## 6 10 8 8  
## cancellation\_policy reviews\_per\_month  
## 1 strict\_14\_with\_grace\_period 4.83  
## 2 strict\_14\_with\_grace\_period 0.03  
## 3 strict\_14\_with\_grace\_period 3.63  
## 4 strict\_14\_with\_grace\_period 0.18  
## 5 strict\_14\_with\_grace\_period 0.64  
## 6 strict\_14\_with\_grace\_period 0.77

names(Airbnb\_Sydney)

## [1] "id" "description"   
## [3] "neighborhood\_overview" "house\_rules"   
## [5] "host\_id" "host\_since"   
## [7] "host\_response\_time" "host\_response\_rate"   
## [9] "host\_is\_superhost" "host\_verifications"   
## [11] "host\_identity\_verified" "city"   
## [13] "zipcode" "property\_type"   
## [15] "room\_type" "accommodates"   
## [17] "bathrooms" "bedrooms"   
## [19] "beds" "bed\_type"   
## [21] "amenities" "price"   
## [23] "cleaning\_fee" "guests\_included"   
## [25] "extra\_people" "minimum\_nights"   
## [27] "number\_of\_reviews" "review\_scores\_rating"   
## [29] "review\_scores\_accuracy" "review\_scores\_cleanliness"   
## [31] "review\_scores\_checkin" "review\_scores\_communication"  
## [33] "review\_scores\_location" "review\_scores\_value"   
## [35] "cancellation\_policy" "reviews\_per\_month"

dim(Airbnb\_Sydney)

## [1] 10815 36

class(Airbnb\_Sydney)

## [1] "data.frame"

anyNA((Airbnb\_Sydney))

## [1] TRUE

sum(is.na.data.frame(Airbnb\_Sydney))

## [1] 7

##### 1.1

I combed through the csv file to look for possible missing values and found several possbilities including the usual NA values. I wrote a function to find and sum up the instances of these missing data for each column. My results show that the neighborhood\_overview, house\_rules, host\_response\_time host\_response\_rate, host\_identity\_verified, city, zipcode, bathrooms, bedrooms, cleaning\_fee, review\_scores\_rating, review\_scores\_accuracy, review\_scores\_cleanliness, review\_scores\_checkin, and review\_scores\_communication columns have a missing values. Those missing values are NA, N/A and empty strings.

missing\_everything = function(x){  
 # this function takes on argument and finds a list of possible missing values and returns a vector with the number of times that missing value occurs  
 z = sum(as.numeric(is.na(x)))  
 y = sum(as.numeric(x==""))  
 x = sum(as.numeric(x=="N/A"))  
 w = sum(as.numeric(x=="[]"))  
 v = sum(as.numeric(x=="æ,%å•1/4"))  
 u = sum(as.numeric(x=="#NAME?"))  
 t = sum(as.numeric(x=="."))  
 s = sum(as.numeric(x=="/"))  
 r = sum(as.numeric(x=="(URL HIDDEN)"))  
 q = sum(as.numeric(x=="(Other)"))  
 return(c(z,y,x,w,v,u,t,s,r,q))  
}  
  
# calling the function using apply()  
apply(Airbnb\_Sydney,2,missing\_everything)

## id description neighborhood\_overview house\_rules host\_id host\_since  
## [1,] 0 0 0 0 0 0  
## [2,] 0 0 664 1639 0 0  
## [3,] 0 0 0 0 0 0  
## [4,] 0 0 0 0 0 0  
## [5,] 0 0 0 0 0 0  
## [6,] 0 0 0 0 0 0  
## [7,] 0 0 0 0 0 0  
## [8,] 0 0 0 0 0 0  
## [9,] 0 0 0 0 0 0  
## [10,] 0 0 0 0 0 0  
## host\_response\_time host\_response\_rate host\_is\_superhost  
## [1,] 0 0 0  
## [2,] 0 0 0  
## [3,] 2483 2483 0  
## [4,] 0 0 0  
## [5,] 0 0 0  
## [6,] 0 0 0  
## [7,] 0 0 0  
## [8,] 0 0 0  
## [9,] 0 0 0  
## [10,] 0 0 0  
## host\_verifications host\_identity\_verified city zipcode property\_type  
## [1,] 0 0 0 0 0  
## [2,] 0 0 8 21 0  
## [3,] 0 0 0 0 0  
## [4,] 0 0 0 0 0  
## [5,] 0 0 0 0 0  
## [6,] 0 0 0 0 0  
## [7,] 0 0 0 0 0  
## [8,] 0 0 0 0 0  
## [9,] 0 0 0 0 0  
## [10,] 0 0 0 0 0  
## room\_type accommodates bathrooms bedrooms beds bed\_type amenities  
## [1,] 0 0 1 1 0 0 0  
## [2,] 0 0 NA NA 0 0 0  
## [3,] 0 0 NA NA 0 0 0  
## [4,] 0 0 NA NA 0 0 0  
## [5,] 0 0 NA NA 0 0 0  
## [6,] 0 0 NA NA 0 0 0  
## [7,] 0 0 NA NA 0 0 0  
## [8,] 0 0 NA NA 0 0 0  
## [9,] 0 0 NA NA 0 0 0  
## [10,] 0 0 NA NA 0 0 0  
## price cleaning\_fee guests\_included extra\_people minimum\_nights  
## [1,] 0 0 0 0 0  
## [2,] 0 621 0 0 0  
## [3,] 0 0 0 0 0  
## [4,] 0 0 0 0 0  
## [5,] 0 0 0 0 0  
## [6,] 0 0 0 0 0  
## [7,] 0 0 0 0 0  
## [8,] 0 0 0 0 0  
## [9,] 0 0 0 0 0  
## [10,] 0 0 0 0 0  
## number\_of\_reviews review\_scores\_rating review\_scores\_accuracy  
## [1,] 0 1 1  
## [2,] 0 NA NA  
## [3,] 0 NA NA  
## [4,] 0 NA NA  
## [5,] 0 NA NA  
## [6,] 0 NA NA  
## [7,] 0 NA NA  
## [8,] 0 NA NA  
## [9,] 0 NA NA  
## [10,] 0 NA NA  
## review\_scores\_cleanliness review\_scores\_checkin  
## [1,] 1 1  
## [2,] NA NA  
## [3,] NA NA  
## [4,] NA NA  
## [5,] NA NA  
## [6,] NA NA  
## [7,] NA NA  
## [8,] NA NA  
## [9,] NA NA  
## [10,] NA NA  
## review\_scores\_communication review\_scores\_location  
## [1,] 1 0  
## [2,] NA 0  
## [3,] NA 0  
## [4,] NA 0  
## [5,] NA 0  
## [6,] NA 0  
## [7,] NA 0  
## [8,] NA 0  
## [9,] NA 0  
## [10,] NA 0  
## review\_scores\_value cancellation\_policy reviews\_per\_month  
## [1,] 0 0 0  
## [2,] 0 0 0  
## [3,] 0 0 0  
## [4,] 0 0 0  
## [5,] 0 0 0  
## [6,] 0 0 0  
## [7,] 0 0 0  
## [8,] 0 0 0  
## [9,] 0 0 0  
## [10,] 0 0 0

##### 1.2

To deal with this, I will be keeping the NA’s in my dataset, but I will be replacing all other missing values with NA. This way I can easily exclude them when needed. I will keep the NA’s in the dataset because I don’t want to exlcude those rows that contain NA’s from the dataset as I want to include that data in my analysis. I feel this is will help me get the most out of the data in terms of insights.

##### 1.3

Handling the missing values this way will mean I will have to be mindful of the type of analysis I’m doing, and decide whether or not I have to explude the NA’s. There will be cases where I have to exclude them to do an operation in R. However, I think my output and report should also reflect the amount of missing data, to help with the interpretation of my results. Knowing that there is missing data (and how much) could influence decisions that are made using this analysis. I think it would bias the results if I did not include the NA values.

##### 1.4

I replaced the missing values “N/A” and “” with NA, to make missing values consistent and easy to exclude later.

# Replacing weird missing values with NA  
Airbnb\_Sydney$host\_response\_time[Airbnb\_Sydney$host\_response\_time=="N/A"] <- NA  
Airbnb\_Sydney$host\_response\_rate[Airbnb\_Sydney$host\_response\_rate=="N/A"] <- NA  
Airbnb\_Sydney$neighborhood\_overview[Airbnb\_Sydney$neighborhood\_overview==""] <- NA  
Airbnb\_Sydney$house\_rules[Airbnb\_Sydney$house\_rules==""] <- NA  
Airbnb\_Sydney$zipcode[Airbnb\_Sydney$zipcode==""] <- NA  
Airbnb\_Sydney$cleaning\_fee[Airbnb\_Sydney$cleaning\_fee==""] <- NA  
  
#apply(Airbnb\_Sydney,2,missing\_everything)  
sum(is.na.data.frame(Airbnb\_Sydney))

## [1] 7918

apply(is.na(Airbnb\_Sydney), 2, sum)

## id description   
## 0 0   
## neighborhood\_overview house\_rules   
## 664 1639   
## host\_id host\_since   
## 0 0   
## host\_response\_time host\_response\_rate   
## 2483 2483   
## host\_is\_superhost host\_verifications   
## 0 0   
## host\_identity\_verified city   
## 0 0   
## zipcode property\_type   
## 21 0   
## room\_type accommodates   
## 0 0   
## bathrooms bedrooms   
## 1 1   
## beds bed\_type   
## 0 0   
## amenities price   
## 0 0   
## cleaning\_fee guests\_included   
## 621 0   
## extra\_people minimum\_nights   
## 0 0   
## number\_of\_reviews review\_scores\_rating   
## 0 1   
## review\_scores\_accuracy review\_scores\_cleanliness   
## 1 1   
## review\_scores\_checkin review\_scores\_communication   
## 1 1   
## review\_scores\_location review\_scores\_value   
## 0 0   
## cancellation\_policy reviews\_per\_month   
## 0 0

##### 1.5

dim(Airbnb\_Sydney)

## [1] 10815 36

# The dim will reamin the same, because I have not removed any NA values yet.

##### 1.6

I will remove any weird characters like dollar signs from (potentially) numeric columns. I will look at all data types. The date columns may need to be formatted, or broke up for more detailed analysis (for instance to look at a single year, month, or day). All classes of the columns should be doubled checked, because they may need to be re-assgined to be an integer from a character, for instance. I would double check that the data was read in correctly with the right headers. To analyze text data, I would parse and remove any strange characters to understand what words or how many words are in a row.

#### Q2

I looked at the overall distribution of the dataframe and decided to focus, at first, the the property type, the city, price and host since columns. I felt these variables would offer the most insight into the dataset.

# The funcitons below were used to help me understand the dataset:  
# summary and str are commented out because output is so large  
#summary(Airbnb\_Sydney)  
#str(Airbnb\_Sydney)  
names(Airbnb\_Sydney)

## [1] "id" "description"   
## [3] "neighborhood\_overview" "house\_rules"   
## [5] "host\_id" "host\_since"   
## [7] "host\_response\_time" "host\_response\_rate"   
## [9] "host\_is\_superhost" "host\_verifications"   
## [11] "host\_identity\_verified" "city"   
## [13] "zipcode" "property\_type"   
## [15] "room\_type" "accommodates"   
## [17] "bathrooms" "bedrooms"   
## [19] "beds" "bed\_type"   
## [21] "amenities" "price"   
## [23] "cleaning\_fee" "guests\_included"   
## [25] "extra\_people" "minimum\_nights"   
## [27] "number\_of\_reviews" "review\_scores\_rating"   
## [29] "review\_scores\_accuracy" "review\_scores\_cleanliness"   
## [31] "review\_scores\_checkin" "review\_scores\_communication"  
## [33] "review\_scores\_location" "review\_scores\_value"   
## [35] "cancellation\_policy" "reviews\_per\_month"

Looking closer at the these columns, I see there are a large number of cities and property types to work with as character and factor variables. I also found an “Other” type in the city variable, which I have decided to leave in for now, but may turn into an NA. However, I don’t think “other” means the same thing as NA, so I will decide on a case by case basis on what to do with that. I also looked at price. I removed the dollar sign from the data so I could look at it more closely. I also looked at the distribution and can see that the majority of the prices are at or below 200 dollars. I also looked at host\_since variable, and we can see that we have data from 2009 to 2018 to work with. I also looked at cleaning fees, super host status, and review scores.

#Airbnb\_Sydney$city  
class(Airbnb\_Sydney$city)

## [1] "factor"

anyNA(Airbnb\_Sydney$city)

## [1] FALSE

head(summary(as.factor(Airbnb\_Sydney$city)))

## Bondi Beach Surry Hills Sydney Manly Darlinghurst   
## 555 500 463 389 373   
## Coogee   
## 272

#Airbnb\_Sydney$property\_type  
class(Airbnb\_Sydney$property\_type)

## [1] "factor"

anyNA(Airbnb\_Sydney$property\_type)

## [1] FALSE

head(summary(as.factor(Airbnb\_Sydney$property\_type)))

## Aparthotel Apartment Bed and breakfast Boat   
## 2 6222 46 8   
## Boutique hotel Bungalow   
## 26 62

#Airbnb\_Sydney$property\_type  
class(Airbnb\_Sydney$host\_since)

## [1] "factor"

anyNA(Airbnb\_Sydney$host\_since)

## [1] FALSE

head(sort(as.Date(Airbnb\_Sydney$host\_since, tryFormats = c("%m/%d/%y")), decreasing = T))

## [1] "2018-11-25" "2018-11-21" "2018-11-21" "2018-11-20" "2018-11-19"  
## [6] "2018-11-19"

tail(sort(as.Date(Airbnb\_Sydney$host\_since, tryFormats = c("%m/%d/%y")), decreasing = T))

## [1] "2009-05-17" "2009-05-14" "2009-05-14" "2009-05-14" "2009-04-20"  
## [6] "2009-04-20"

#Airbnb\_Sydney$price  
anyNA(Airbnb\_Sydney$price)

## [1] FALSE

class(Airbnb\_Sydney$price)

## [1] "factor"

price\_num = as.numeric(gsub("[\\$]", "", Airbnb\_Sydney$price), length(2))

## Warning: NAs introduced by coercion

head(summary(price\_num))

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000 96.0000 150.0000 189.3062 226.0000 999.0000

#Airbnb\_Sydney$cleaning\_fee  
anyNA(Airbnb\_Sydney$cleaning\_fee)

## [1] TRUE

class(Airbnb\_Sydney$cleaning\_fee)

## [1] "factor"

clean\_fee = as.numeric(gsub("[\\$]", "", Airbnb\_Sydney$cleaning\_fee), length(2))   
head(summary(clean\_fee))

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.00000 40.00000 80.00000 94.47234 125.00000 800.00000

I also looked at the different review variables and super host variables. I thought these variables would also offer insight, and possbility overlap. Upon further inspection, I can see that the distribution of the review\_rating and review\_value columns are very similar (just on a different scale). Reviews\_per\_month looked interesting, but actually didn’t hold much data, and we can see the distribution is really skewed. The superhost variable did not offer much in terms of insight at this stage, and is a collection of true and false values.

In general, it looks like the data from most variables is skewed, which will be something to keep in mind for analysis.

# Looking at some of the different "review" columns to see which my be useful later and the superhost column  
  
#Airbnb\_Sydney$review\_scores\_value  
class(Airbnb\_Sydney$review\_scores\_value)

## [1] "integer"

summary(Airbnb\_Sydney$review\_scores\_value)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 2.000 9.000 10.000 9.385 10.000 10.000

#Airbnb\_Sydney$review\_scores\_rating  
class(Airbnb\_Sydney$review\_scores\_rating)

## [1] "integer"

summary(Airbnb\_Sydney$review\_scores\_rating)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 20.00 92.00 96.00 94.19 100.00 100.00 1

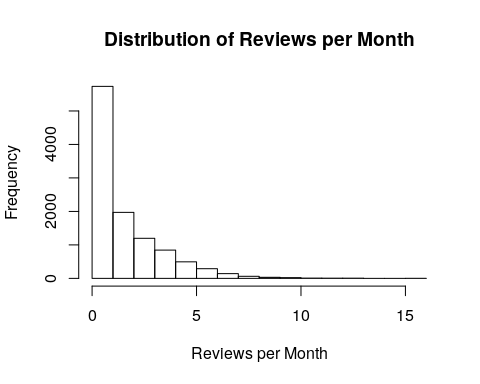
#Airbnb\_Sydney$reviews\_per\_month  
class(Airbnb\_Sydney$reviews\_per\_month)

## [1] "numeric"

summary(Airbnb\_Sydney$reviews\_per\_month)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.020 0.270 0.950 1.572 2.310 15.180

hist(Airbnb\_Sydney$reviews\_per\_month, main = "Distribution of Reviews per Month", xlab = "Reviews per Month")



#Airbnb\_Sydney$host\_is\_superhost  
class(Airbnb\_Sydney$host\_is\_superhost)

## [1] "factor"

summary(Airbnb\_Sydney$host\_is\_superhost)

## f t   
## 8020 2795

head(Airbnb\_Sydney$host\_is\_superhost)

## [1] f f f t f f  
## Levels: f t

#### Q3

##### 3.1

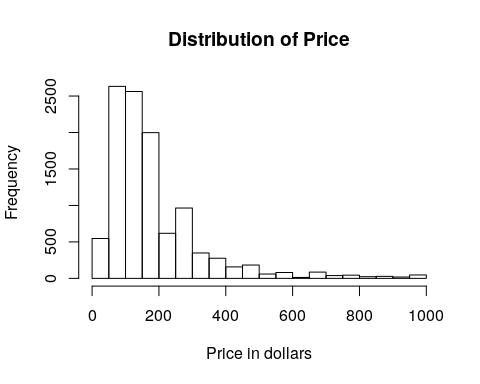
We have mostly character vectors, which I will change to integer, dates and factors for visualization and analysis. Of the possible numerical variables, it looks like we have only discrete variables. To visualize this, I will use scatter plots, barplots, boxplots, and denisty plots.

# what types of data are there?  
apply(Airbnb\_Sydney, 2, class)

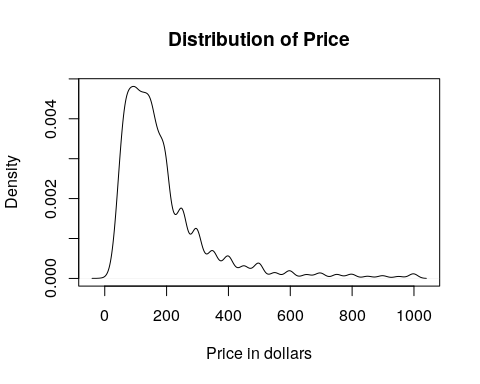
## id description   
## "character" "character"   
## neighborhood\_overview house\_rules   
## "character" "character"   
## host\_id host\_since   
## "character" "character"   
## host\_response\_time host\_response\_rate   
## "character" "character"   
## host\_is\_superhost host\_verifications   
## "character" "character"   
## host\_identity\_verified city   
## "character" "character"   
## zipcode property\_type   
## "character" "character"   
## room\_type accommodates   
## "character" "character"   
## bathrooms bedrooms   
## "character" "character"   
## beds bed\_type   
## "character" "character"   
## amenities price   
## "character" "character"   
## cleaning\_fee guests\_included   
## "character" "character"   
## extra\_people minimum\_nights   
## "character" "character"   
## number\_of\_reviews review\_scores\_rating   
## "character" "character"   
## review\_scores\_accuracy review\_scores\_cleanliness   
## "character" "character"   
## review\_scores\_checkin review\_scores\_communication   
## "character" "character"   
## review\_scores\_location review\_scores\_value   
## "character" "character"   
## cancellation\_policy reviews\_per\_month   
## "character" "character"

##### 3.2

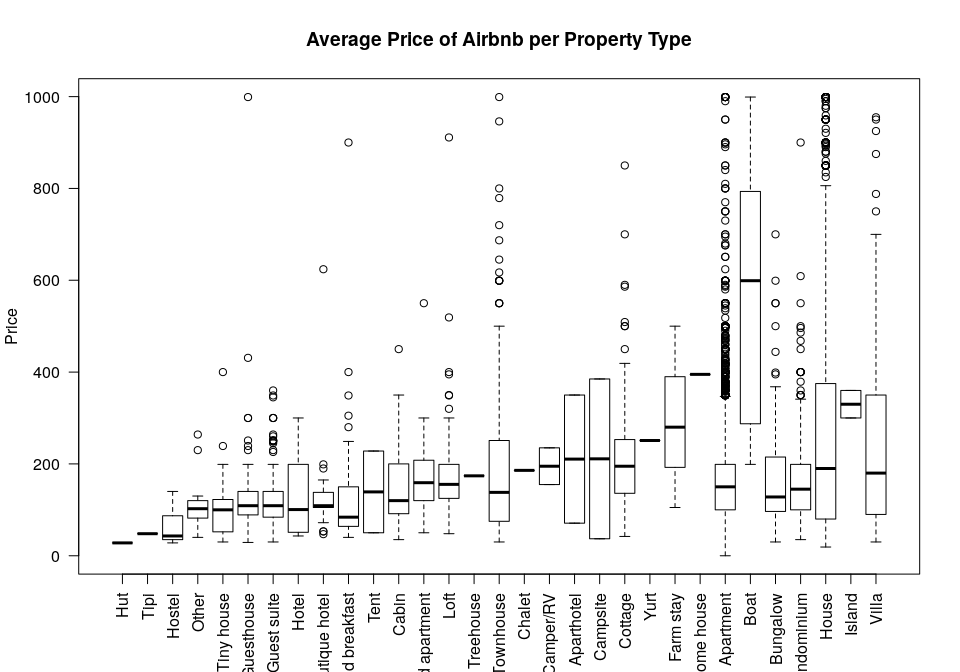
# exploring the price variable using histograms and density plots  
  
hist(price\_num, main = "Distribution of Price", xlab = "Price in dollars")



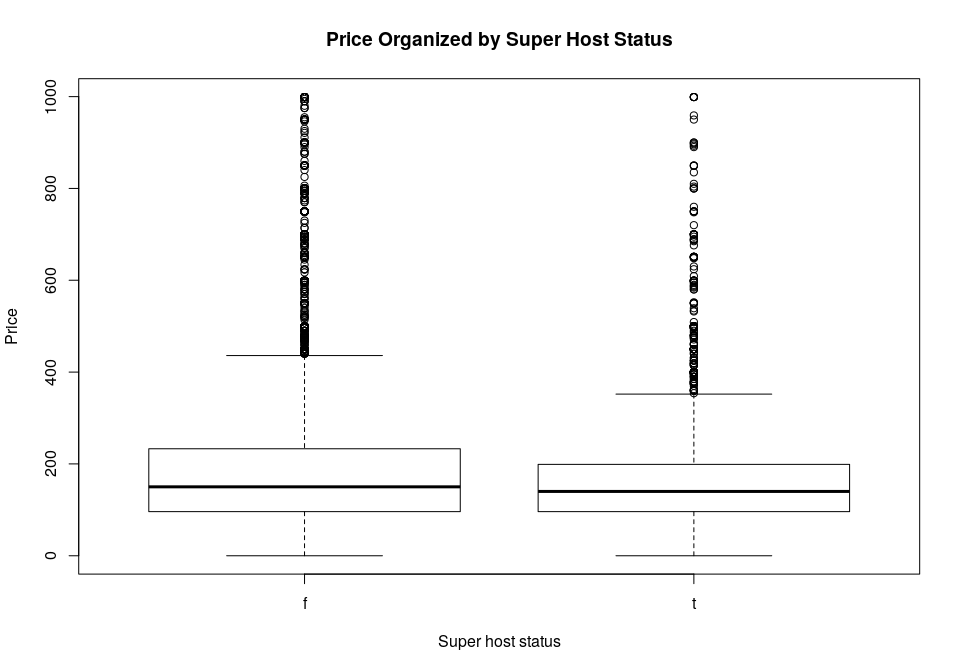
plot(density(price\_num, na.rm = T), main = "Distribution of Price", xlab = "Price in dollars")



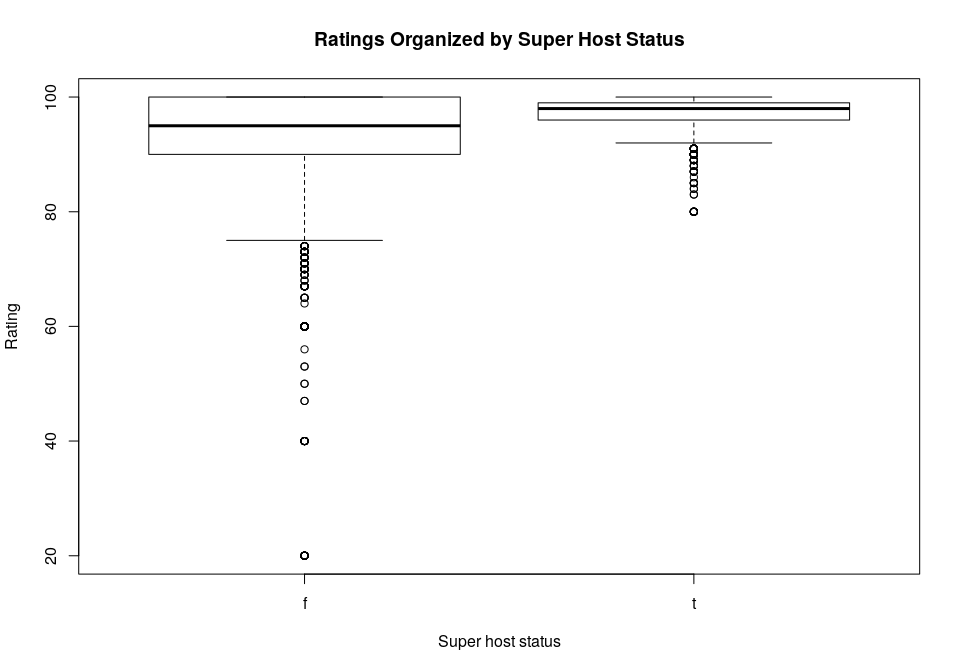
# looking at price and other variables  
# looking at review columns and super host  
  
boxplot(price\_num~  
 as.factor(Airbnb\_Sydney$property\_type),   
 at = rank(tapply(price\_num, Airbnb\_Sydney$property\_type, mean)), las =2,   
 main = "Average Price of Airbnb per Property Type",  
 ylab = "Price", cex.names=0.4)



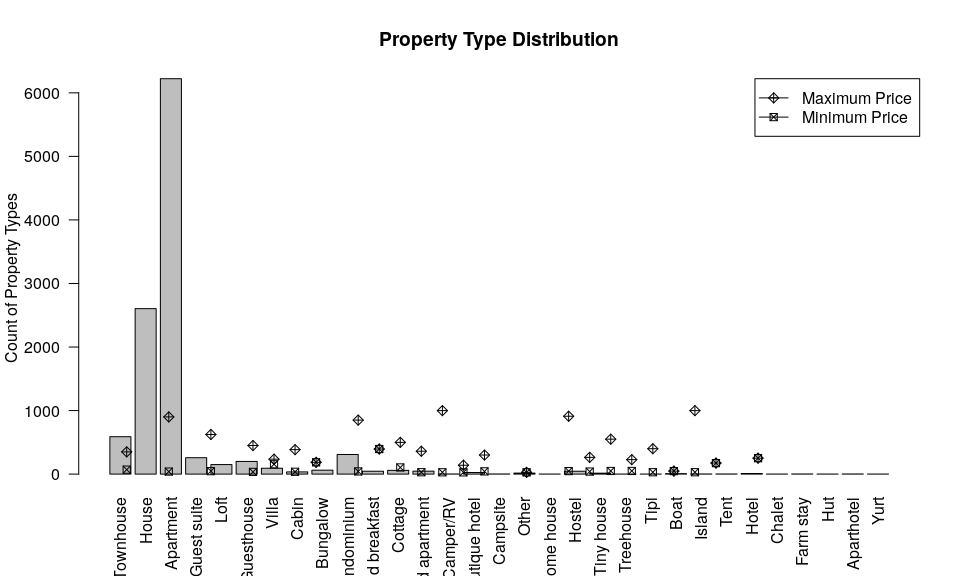
plot(price\_num~as.factor(Airbnb\_Sydney$host\_is\_superhost), main = "Price Organized by Super Host Status",  
 xlab = "Super host status", ylab = "Price")



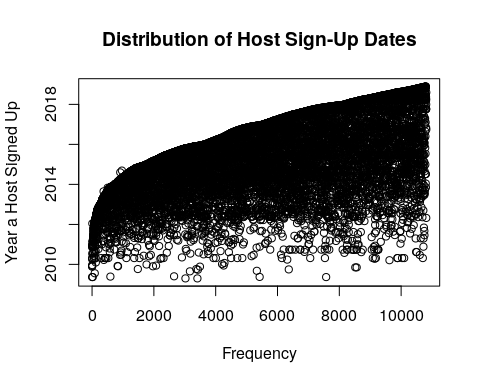
plot(Airbnb\_Sydney$review\_scores\_rating~as.factor(Airbnb\_Sydney$host\_is\_superhost), main = "Ratings Organized by Super Host Status",  
 xlab = "Super host status", ylab = "Rating")



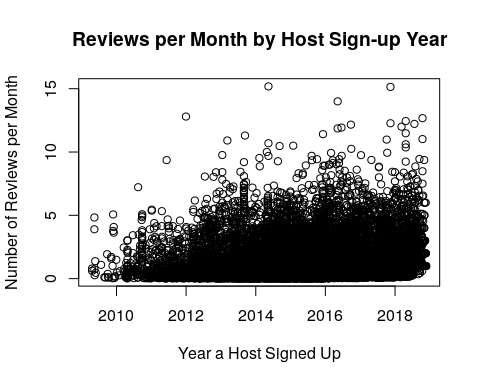
# looking at property type distribution with maximum and minimum plotted over top  
  
plot(sort(factor(Airbnb\_Sydney$property\_type,   
 levels = unique(Airbnb\_Sydney$property\_type)), decreasing = T), las=2, main = "Property Type Distribution", ylab = "Count of Property Types")  
  
onemax <- by(price\_num, Airbnb\_Sydney$property\_type, max)  
means <- by(price\_num, Airbnb\_Sydney$property\_type, min)  
  
points(onemax, pch=9)  
points(means, pch=7)  
legend(legend = c("Maximum Price", "Minimum Price"), lty = c(1, 1), pch = c(9, 7), "topright")



# looking at host since dates and reviews  
  
plot(as.Date(Airbnb\_Sydney$host\_since, tryFormats = c("%m/%d/%y")),   
 main = "Distribution of Host Sign-Up Dates", xlab = "Frequency",  
 ylab = "Year a Host Signed Up")



plot(Airbnb\_Sydney$reviews\_per\_month~as.Date(Airbnb\_Sydney$host\_since, tryFormats = c("%m/%d/%y")),  
 main = "Reviews per Month by Host Sign-up Year", xlab = "Year a Host Signed Up",   
 ylab = "Number of Reviews per Month")



##### 3.3

Visualizing the data showed me the distribution and potential relationship between the variables: price (reamed price\_num), super\_host, property\_type, host\_since, and review\_per\_month.

First, I focused on the price variable, which I converted to a numeric data type. I plotted it as both a histogram and a density plot. The density plot shows the overall trend better, because I have a smooth curve that that shows two peaks close together, and the sharp drop in the number of properties listed for more than 200 dollars a night. The histrogrm shows me more details, however, and I can see the individual frequencies for each category. For exploring the data, the histogram is better. As a presentation tool, the denisty plot is better.

Second, I looked at price in relation to property type using boxplots. I plotted the property type and ranked the properites according to the average price. Here we can see the trend in mean price, and also the amount of variance in price for each property type. Here is some data that will come in handy in the analysis phase. Property types that stand out are apartments, boats, houses, and townhomes. While some properites had massive variance, others did not, such as islands and cottages. Further analysis is needed here. I also organized price by super host status, but nothing interesting appeared at this initial stage, and may it require including other variables to see if super host status influences the data.

To get a better idea about what is going on with property types, I plotted a barchart to get a look at the distribution, and plotted the mean price overtop. So the same data, but looking at it in a different way. Here we can see that there are many apartment listed, followed by houses and townhomes. Here it makes sense that their is variance in the properities listed the most. It also looks like some averages are missing for some property types. The data looks to be skewed in favor of apartments, with little or no data for the majority of the property types.

I then looked at the distribution of host sign-up dates, and we can a drastic increase in hosts from 2010 to 2014. I also looked at the number of reviews organized by host sign up date, and we can see that the data is messy, but there is a general trend where there are more reviews for more recent dates. More analysis is needed here.

#### Q4

##### 4.1

The number of reviews is the total number of reviews for listing, so we can use this varialbe to infer how often a listing has been rented and reviewed and how long it’s been listed. This is useful for looking at a property’s overall business. The number of reviews per month gives us more information on how often a listing was rented at time intervals. We can see how busy a property is or isn’t and look for peak times. These variables are similar in that they convey information about freauency of use, but one gives us a total and one gives us a number for a short period of time.

From the analysis below, it looks like there are 25 host ids that are in the top 100 listings for number of reviews and for number of reviews per month.

# create a dataframe for reviews and host id  
id\_num = Airbnb\_Sydney[,c("host\_id", "number\_of\_reviews", "reviews\_per\_month")]  
  
# subset and order by number of reviews  
dis\_num = id\_num[order(id\_num$number\_of\_reviews, decreasing = T), ]  
top\_total = head(dis\_num, n=100L)  
  
# subset and order by reviews per month  
dis1\_num = id\_num[order(id\_num$reviews\_per\_month, decreasing = T), ]  
top\_monthly = head(dis1\_num, n=100L)  
  
# check for overlapping host ids  
top\_monthly$host\_id %in% top\_total$host\_id

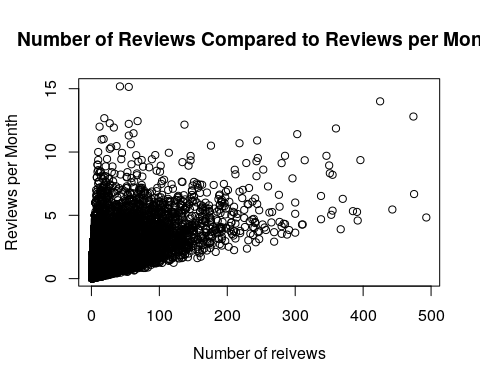
## [1] TRUE FALSE TRUE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## [12] TRUE FALSE TRUE FALSE FALSE FALSE TRUE TRUE FALSE FALSE FALSE  
## [23] FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE TRUE TRUE  
## [34] FALSE FALSE FALSE FALSE FALSE TRUE TRUE FALSE TRUE FALSE TRUE  
## [45] TRUE TRUE FALSE FALSE FALSE TRUE FALSE FALSE FALSE FALSE FALSE  
## [56] FALSE TRUE TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## [67] FALSE TRUE FALSE FALSE TRUE FALSE TRUE FALSE FALSE TRUE FALSE  
## [78] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE FALSE FALSE  
## [89] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  
## [100] FALSE

sum(as.numeric(top\_monthly$host\_id %in% top\_total$host\_id))

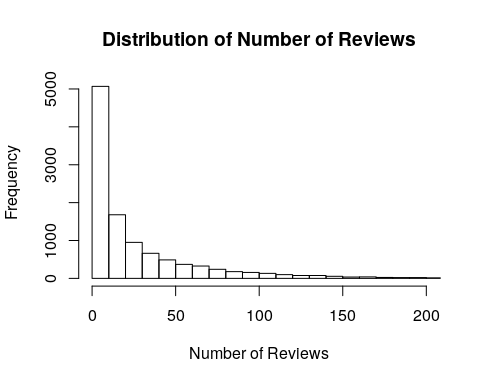
## [1] 25

Taking a look at this relationship visually, we can see that the two variables are closely related, with a higher number overall reviews being correlated with a higher number of reviews per month. This does suggest some colinearity between the two variables. I also looked at the distriubtion of both variables, and they look similar.

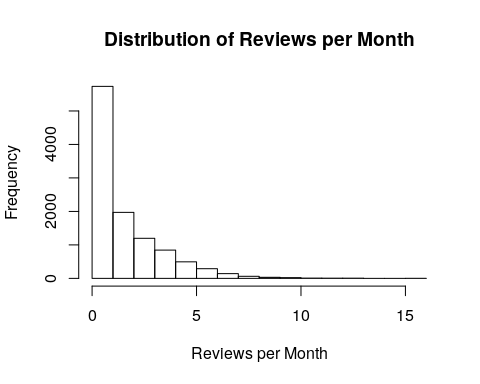
plot(Airbnb\_Sydney$reviews\_per\_month~Airbnb\_Sydney$number\_of\_reviews,  
 main = "Number of Reviews Compared to Reviews per Month", xlab = "Number of reivews",  
 ylab = "Reviews per Month")



hist(Airbnb\_Sydney$number\_of\_reviews, xlim = c(0,200), breaks = 50,   
 main = "Distribution of Number of Reviews", xlab = "Number of Reviews")



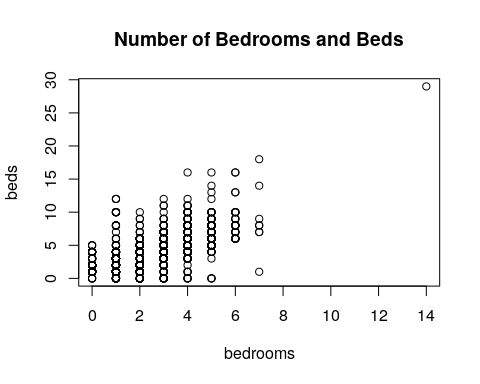
hist(Airbnb\_Sydney$reviews\_per\_month,   
 main = "Distribution of Reviews per Month", xlab = "Reviews per Month")



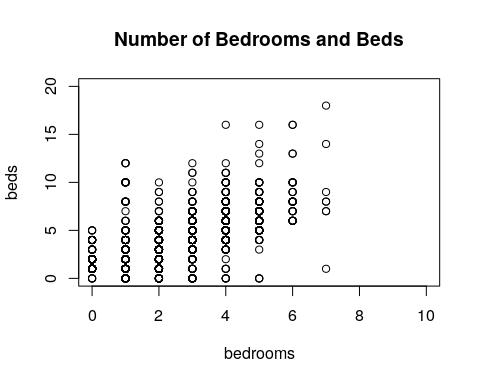
##### 4.2

I also looked at the number of bedrooms and the number of beds in terms of a relationship. We can see that with an increase in bedrooms there is also an increase in the number of beds. This implies that their is a relationship between these two variables and possible colinearity. From both of these variables, we can infer the size of the property and how many peole can stay at this listing.

# number of bedrooms to number of beds  
  
plot(Airbnb\_Sydney[order(Airbnb\_Sydney$bedrooms, decreasing = T),   
 c("bedrooms", "beds")], type = "p", main = "Number of Bedrooms and Beds")



# removed outliers  
  
plot(Airbnb\_Sydney[order(Airbnb\_Sydney$bedrooms, decreasing = T),   
 c("bedrooms", "beds")], type = "p", main = "Number of Bedrooms and Beds",  
 xlim = c(0,10), ylim = c(0,20))



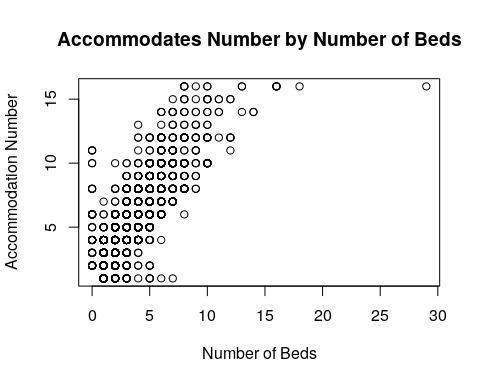
Next, I looked at zipcode and cities, which do have considerable overlap. We can see from the dataframe that I created that each zipcode corresponds to one or two cities. We can also see the reverse when ordering by city. This means that we only need one of these variables in analysis, as they both represent similar data.

ord\_zip = Airbnb\_Sydney[order(Airbnb\_Sydney$number\_of\_reviews, decreasing = T), c("host\_id","zipcode","city","number\_of\_reviews")]  
  
top\_zip = head(ord\_zip, n=100L)  
top\_zip

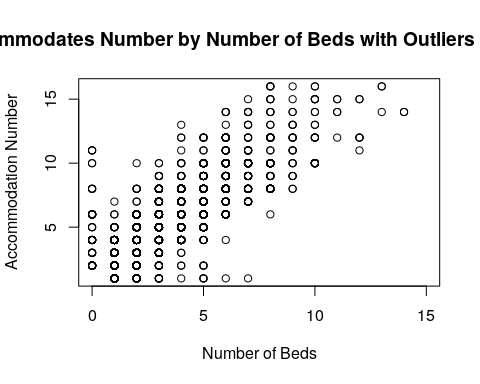
## host\_id zipcode city number\_of\_reviews  
## 1 17061 2009 Pyrmont 493  
## 298 4798499 2107 Avalon 475  
## 2703 1553030 2020 Mascot 474  
## 100 1943399 2016 Redfern 443  
## 4241 71193770 2205 Arncliffe 425  
## 2074 688781 2042 Newtown 396  
## 57 1347315 2026 Bondi 392  
## 183 1943399 <NA> Redfern 391  
## 262 4421400 2010 Surry Hills 385  
## 780 3191055 2011 Woolloomooloo 370  
## 22 17061 2009 Pyrmont 367  
## 4307 71193770 2205 Arncliffe 360  
## 285 4421400 2010 Darlinghurst 355  
## 2228 32928357 2000 Millers Point 355  
## 320 5169464 2016 Redfern 353  
## 2087 30075514 2041 Balmain 351  
## 2839 35454201 2107 Avalon Beach 350  
## 3594 41011048 2020 Mascot 346  
## 198 3720786 2000 Millers Point 338  
## 1009 15354654 2010 Sydney 338  
## 3932 41498403 2044 Tempe 314  
## 210 3809995 2011 Sydney 311  
## 247 4367105 2042 Newtown 310  
## 4777 50707354 2038 Annandale 303  
## 3 59850 2010 Darlinghurst 300  
## 819 4421400 2010 Darlinghurst 300  
## 1443 7531199 2016 Redfern 300  
## 3191 48538672 2034 Coogee 296  
## 170 3437247 2010 Darlinghurst 291  
## 76 209719 2010 Darlinghurst 288  
## 393 6403880 2061 Milsons Point 285  
## 4383 78920515 2206 Earlwood 285  
## 440 4234278 2203 Dulwich Hill 283  
## 105 2020595 2100 Allambie Heights 280  
## 375 410489 2033 Kensington 280  
## 4294 73662200 2044 Tempe 280  
## 554 8939894 2061 Kirribilli 277  
## 1610 4421400 2010 Surry Hills 277  
## 2378 23817711 2011 Woolloomooloo 276  
## 26 425305 2011 Sydney 269  
## 130 2351093 2000 Sydney 269  
## 1205 6248691 2021 Paddington 266  
## 592 3809995 2011 Potts Point 265  
## 1263 18495707 2000 Sydney 262  
## 3549 6344154 2061 Kirribilli 260  
## 404 3455633 2042 Newtown 257  
## 418 4424088 2011 Potts Point 255  
## 4377 74564218 2000 Sydney 253  
## 495 8247293 2011 Potts Point 252  
## 119 2157195 2010 Surry Hills 247  
## 405 6538998 2016 Redfern 247  
## 1960 28356414 2229 Caringbah South 247  
## 379 3792649 2026 Bondi Beach 245  
## 2692 12060025 2095 Manly 245  
## 2474 11889319 2011 Potts Point 244  
## 4841 5383558 2016 Redfern 244  
## 4356 8530753 2000 Sydney 243  
## 4757 20381574 2016 Surry Hills 243  
## 66 1261500 2011 Potts Point 242  
## 1015 15492831 2024 Bronte 241  
## 1019 371323 2026 Bondi Beach 241  
## 3026 46909702 2007 Ultimo 241  
## 458 4155754 2011 Potts Point 240  
## 1320 2255025 2008 Darlington 239  
## 330 5512046 2010 Darlinghurst 238  
## 827 59850 2010 Darlinghurst 238  
## 1335 4945327 2007 Ultimo 237  
## 2761 13767099 2062 Cammeray 237  
## 428 7028222 2011 Potts Point 234  
## 1083 4519063 2068 Willoughby 233  
## 4033 8530753 2000 Sydney 232  
## 154 1261500 2011 Potts Point 231  
## 2531 38607216 2104 Bayview 231  
## 2865 16477385 2035 Maroubra 230  
## 10 279955 2088 Mosman 229  
## 4969 63558864 2217 Kogarah 228  
## 985 1560268 2010 Surry Hills 226  
## 1786 7736332 2042 Newtown 225  
## 2150 15885982 2011 Woolloomooloo 225  
## 4121 60423487 2037 Forest Lodge 223  
## 2917 45620575 2042 Newtown 222  
## 585 3699017 2024 Waverley 219  
## 840 57949 2010 Surry Hills 219  
## 4037 8530753 2000 Sydney 218  
## 6002 15542638 2017 Waterloo 218  
## 2375 19315857 2207 Bexley North 215  
## 2116 20493747 2010 Surry Hills 214  
## 792 11247892 2016 Redfern 213  
## 381 6245401 2230 Bundeena 212  
## 1649 18533922 2044 St Peters 212  
## 4870 74564218 2000 Sydney 212  
## 5215 105151106 2011 Potts Point 211  
## 24 402292 2036 Malabar 210  
## 924 13059157 2010 Darlinghurst 210  
## 2526 7058720 2026 Bondi Beach 210  
## 3016 10859587 2205 Wolli Creek 210  
## 605 9972513 2035 Maroubra 209  
## 2766 6599322 2010 Surry Hills 208  
## 767 11282313 2010 Darlinghurst 207  
## 1854 26709417 2011 Rushcutters Bay 206

I also looked at the relationship between the number of beds and the number people a listing would accommodate. We can see that there is an increase in the number of beds and the number of people allowed to stay. It’s not quite a linear looking relationship, but it is enough overlap to suggest the two are related.

# number of beds and accommodation numbers  
plot(Airbnb\_Sydney$accommodates~Airbnb\_Sydney$beds, type ="p",   
 main = "Accommodates Number by Number of Beds", xlab = "Number of Beds",  
ylab = "Accommodation Number")



# remove outliers  
plot(Airbnb\_Sydney$accommodates~Airbnb\_Sydney$beds, type ="p", xlim= c(0,15), main = "Accommodates Number by Number of Beds with Outliers Removed", xlab = "Number of Beds",  
ylab = "Accommodation Number")



#### Q5

1. I think that there is an optimal price range to attract a high volumne of customers. Listings that fall into this range will have more overall reviews, which will be measured by the number of reviews per month and total reviews. I will start by looking the range and distribution of overall price, and then look at that distriubtion by location (city), because I think that optimal price range will depend on the location. I will then group the prices into bins, and look at how many reviews per month fall into each price range bin. I will include other variable in this analysis such as property type, number of people staying there, and location.
2. I think that hosts who have been using Aribnb longer will have higher review ratings. I will use the host\_since date variable, number of reviews, and ratings to investigate this. I think super host status will need to be held constant, as well as price, location, property type, and beds.
3. I think that a high cleaning fee will lead to lower review ratings and less customers. I will start this analysis by looking at the distribution of the cleaning fee, and then plot that as function of the total reviews and reviews per month, and reviews on cleanliness. I will include other functions such as property type, price, location and number of people staying that I think will influence the analysis. I think price and property type in particular will need to be held constant. So I would look at cleaning fees for apartments or houses only to and number of reviews to understand this relationship.

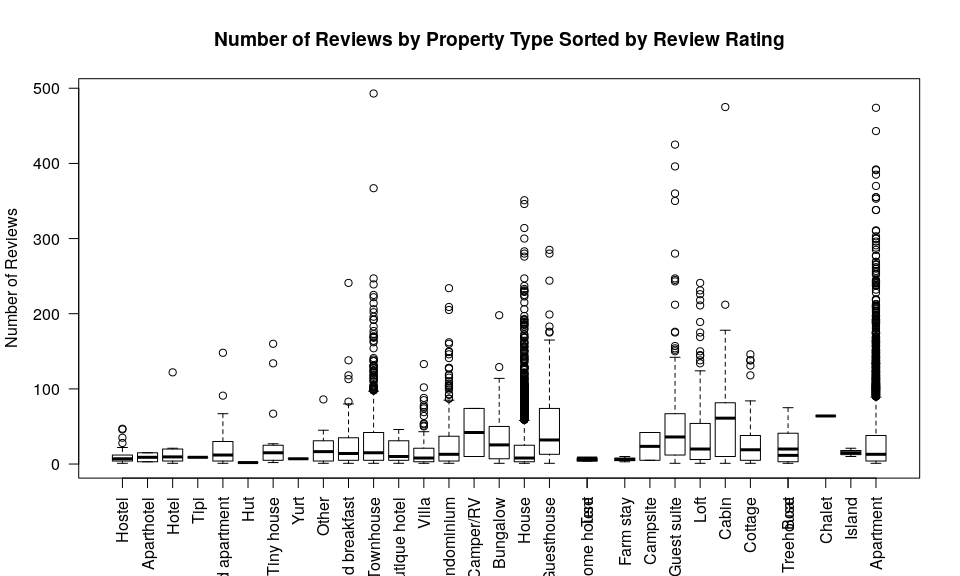
## Part 2

#### Q6

##### 6.1

Below, I have sorted the number of reviews by property type, and sorted them by their average review rating. We can see that apartments have the highest average reivews, and most total number of reivews. After that, we can see that islands and chalet’s also have high average reviews, but not very many total reviews. Apartments have the most reviews overall, but not the highest average. Most of the property types in the data are apartments, so this makes sense.

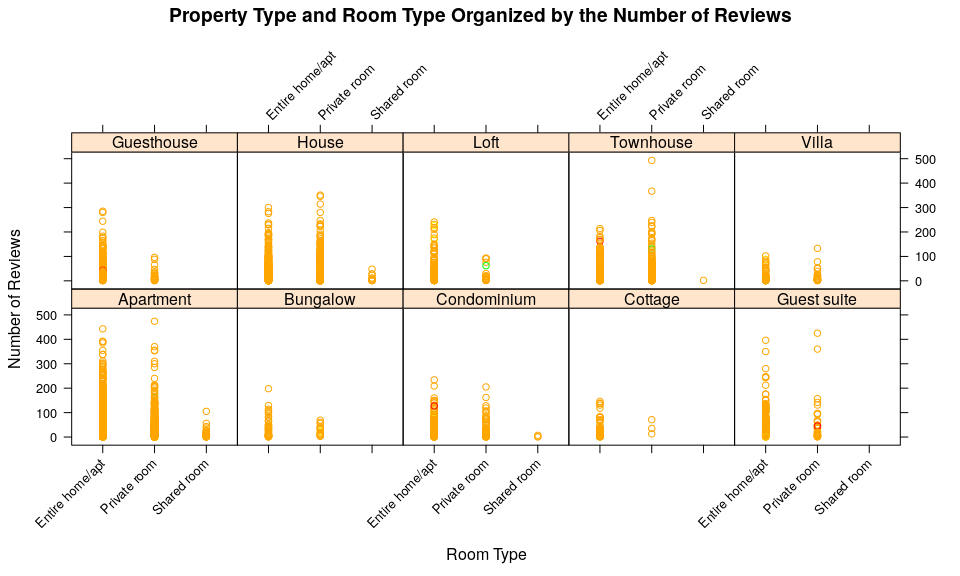
boxplot(Airbnb\_Sydney$number\_of\_reviews~  
 as.factor(Airbnb\_Sydney$property\_type),   
 at = rank(tapply(Airbnb\_Sydney$review\_scores\_rating,   
 Airbnb\_Sydney$property\_type, mean)), las =2,   
 main = "Number of Reviews by Property Type Sorted by Review Rating",  
 ylab = "Number of Reviews")



##### 6.2

From the graph below, we can see that the room types with the highest total reviews for the top ten property types go to the entire home, or some cases just the private room. Shared rooms (across property types) do not have as many total reviews, or any at all. People seem to review the entire listing experience more than they do just a private room or shared room experience. In terms of the bed types, we can see that most of these listings have real beds (shown in orange), with a few execptions.

library(lattice)  
  
# create table  
pnum = as.data.frame(table(Airbnb\_Sydney$property\_type))  
  
# order table  
rpnum = pnum[order(-pnum$Freq), ]  
  
# subset for the top 10 property types  
order\_prop = as.data.frame(Airbnb\_Sydney[c(Airbnb\_Sydney$property\_type=="Apartment"   
 | Airbnb\_Sydney$property\_type=="House"   
 | Airbnb\_Sydney$property\_type=="Townhouse"   
 | Airbnb\_Sydney$property\_type=="Condominium"   
 | Airbnb\_Sydney$property\_type=="Guest suite"   
 | Airbnb\_Sydney$property\_type=="Guesthouse"   
 | Airbnb\_Sydney$property\_type=="Loft"   
 | Airbnb\_Sydney$property\_type=="Villa"   
 | Airbnb\_Sydney$property\_type=="Bungalow"   
 | Airbnb\_Sydney$property\_type=="Cottage"),  
 c("property\_type", "number\_of\_reviews", "room\_type", "bed\_type")])  
  
  
# create factors from bed numbers  
bed\_col = cut(as.numeric(order\_prop$bed\_type), breaks = 5)  
col\_test = c("red", "blue", "green", "yellow", "orange")  
  
# subset bed categories and colors  
bed\_color <- col\_test[bed\_col]  
  
# plot everything  
xyplot(number\_of\_reviews~order\_prop$room\_type|property\_type, data = order\_prop,   
 col=as.character(bed\_color), scales=list(x=list(rot=45)),   
 main = "Property Type and Room Type Organized by the Number of Reviews", xlab = "Room Type", ylab =  
 "Number of Reviews",   
 auto.key = T)

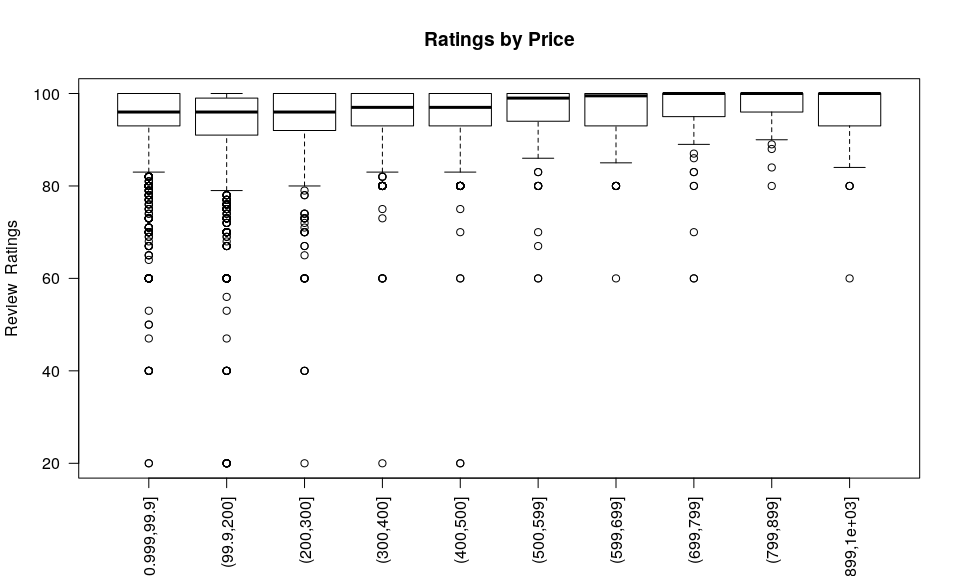


##### 6.3

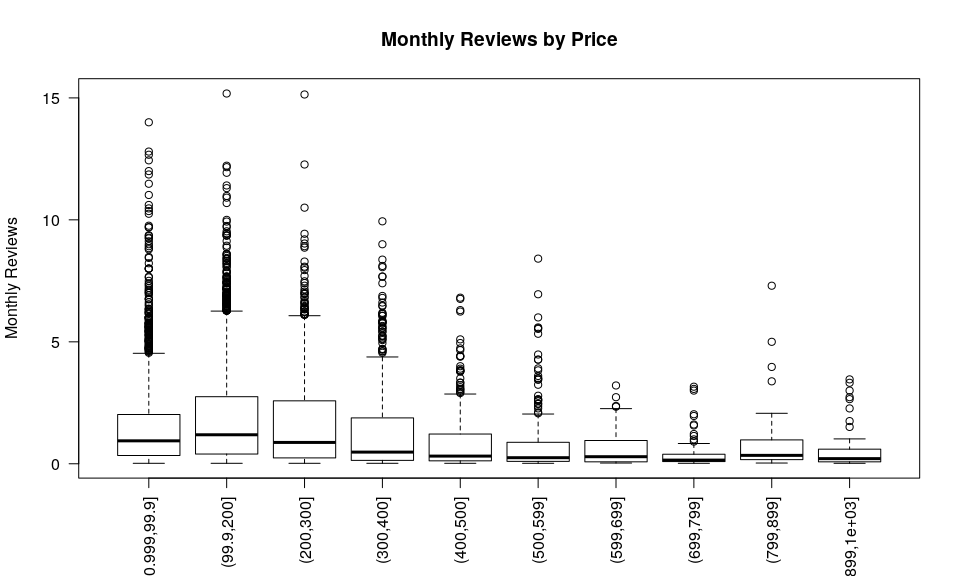
To explore my first hypothesis, I used boxplots to explore how price is related the number of reviews for a listing. I hypothesized that their is an optimal price range to get the most possible reviews. I created categories for price ranges, and plotted the number of reviews, review ratings, and monthly reviews in these categories.

From the visuals it is clear that the number of reviews vary the most (from 10 to less than 20) when the price is lower. Higher priced listings have a smaller range of ratings, as well as an average that is closer to 10. This could also be because higher priced listings are rented less (there’s less data to work with). To further investigate this, we can look at the next graph, also a boxplot, and see that cheaper properties have more ratings per month, which could indicate that cheaper properties are rented more often. However, properties are are in the 100-300 range get the most reviews per month. This is also supported in the next graph, where we can see the total number of reviews are sorted by price category, and we see a similar pattern for the 100-200 dollar category. I used boxplots to show the distributions between different categories for easy comparison.

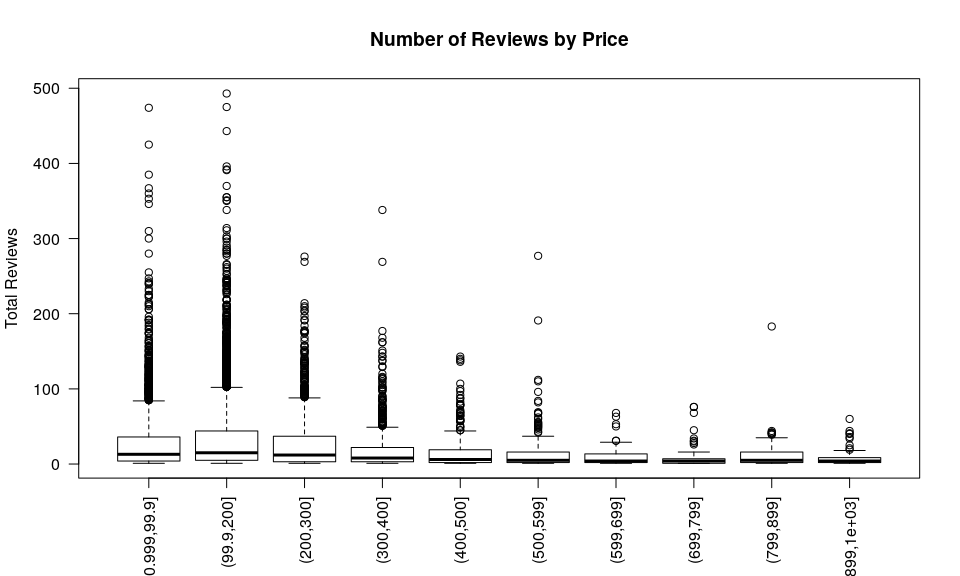
price\_factor <- cut(x = as.numeric(price\_num), breaks = 10)  
  
plot(Airbnb\_Sydney$review\_scores\_rating~price\_factor, las=2,  
 main = "Ratings by Price", xlab = "", ylab = "Review Ratings")



plot(Airbnb\_Sydney$reviews\_per\_month~price\_factor, las= 2,  
 main = "Monthly Reviews by Price", xlab = "", ylab = "Monthly Reviews")



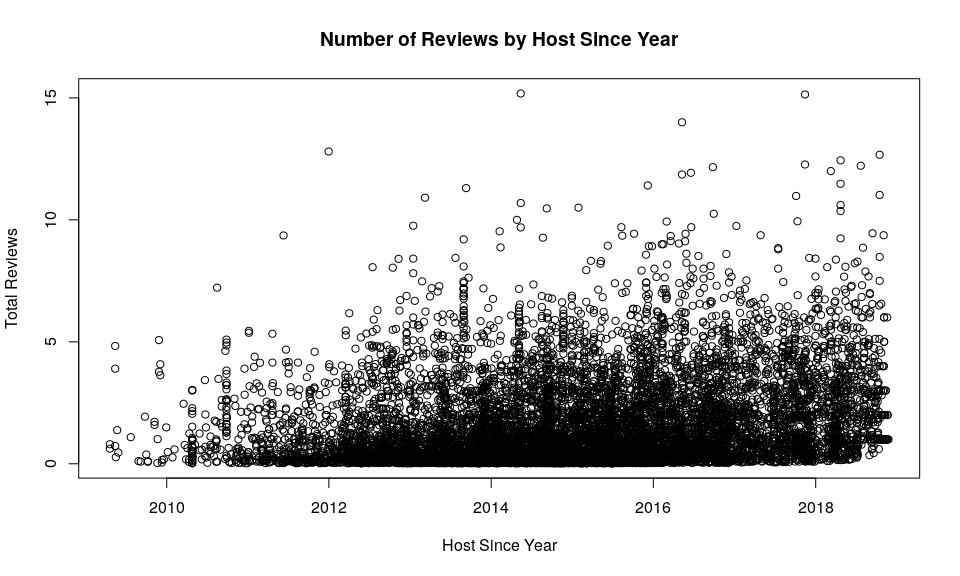
plot(Airbnb\_Sydney$number\_of\_reviews~price\_factor, las=2,  
 main = "Number of Reviews by Price", xlab = "", ylab = "Total Reviews")



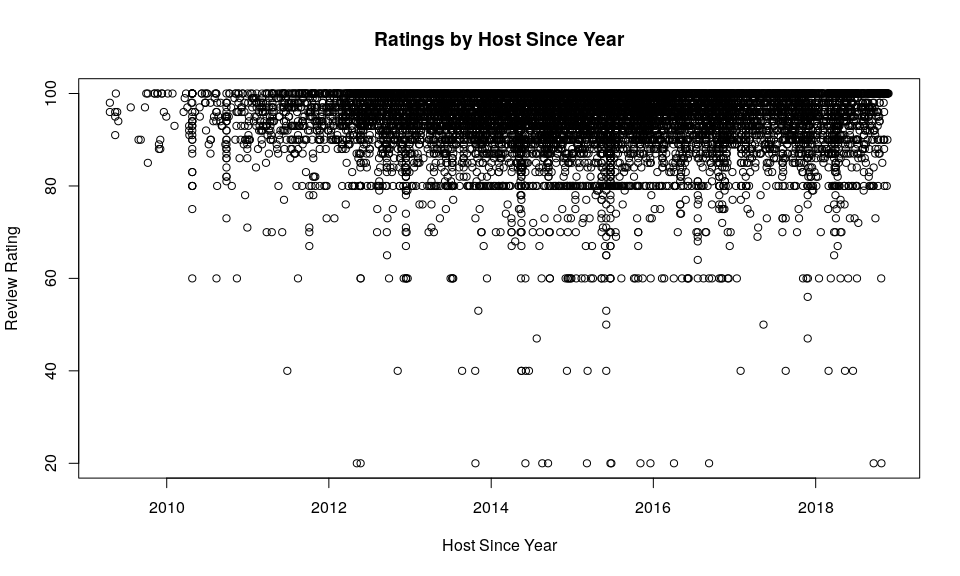
To explore my second hypothesis, I used scatterplots to explore how host sign up date is related the number of bookings and review ratings for a listing. I hypothesized that hosts how have used Aribnb longer will have better ratings.

The first and second visuals shows us that number of reviews and ratings grow with more recent host sign up dates, with the range of ratings increasing as more hosts sign up. The third visual shows host sign up dates by price, with the color indicating the number of beds in a listing. We can see that price tends to go up with the number of beds, and that listings with more beds have appeared with more recent listings. I used scatterplots to show the relationship between ratings, number of reviews, and price to host sign up.

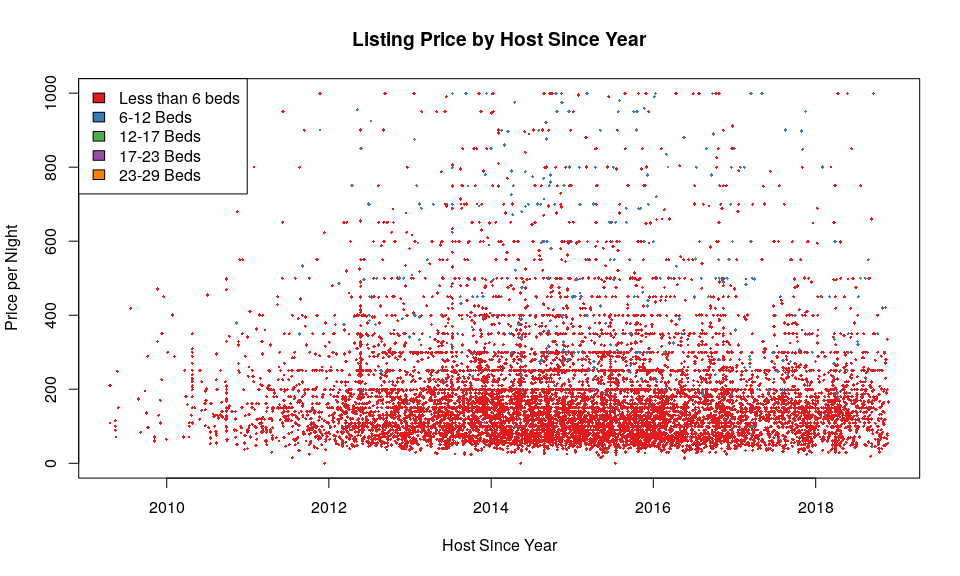
library("RColorBrewer")  
  
Airbnb\_Sydney$host\_data = as.Date(Airbnb\_Sydney$host\_since, tryFormats = c("%m/%d/%y"))  
  
plot(Airbnb\_Sydney$reviews\_per\_month~as.Date(Airbnb\_Sydney$host\_since, tryFormats = c("%m/%d/%y")), main = "Number of Reviews by Host Since Year",  
 xlab = "Host Since Year", ylab = "Total Reviews")



plot(Airbnb\_Sydney$review\_scores\_rating~as.Date(Airbnb\_Sydney$host\_since, tryFormats = c("%m/%d/%y")), main = "Ratings by Host Since Year",  
 xlab = "Host Since Year", ylab = "Review Rating")



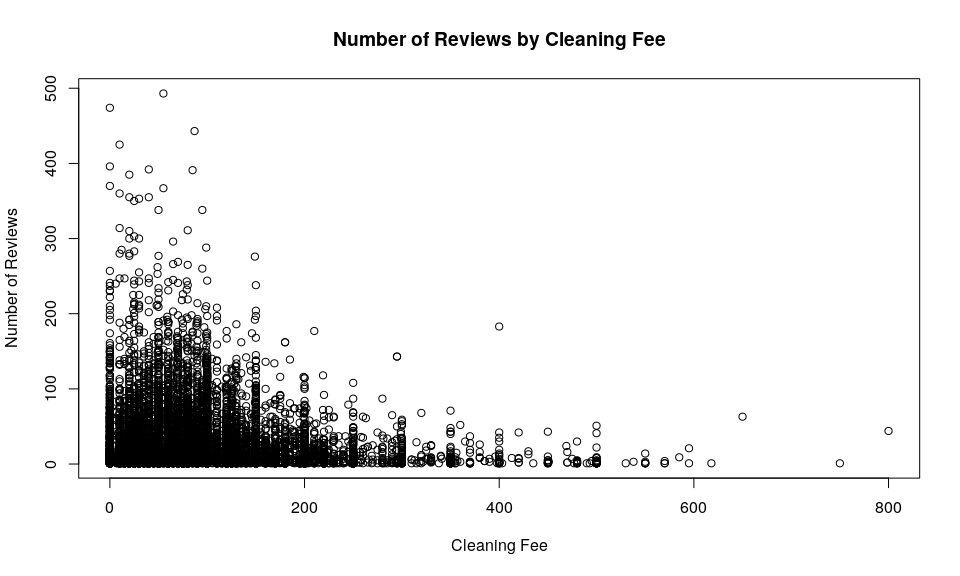
beds\_col = cut(as.numeric(Airbnb\_Sydney$beds), breaks = 5, dig.lab = 1)  
rCols <- brewer.pal(5, name = "Set1")  
brCols <- rCols[beds\_col]  
  
plot(price\_num~as.Date(Airbnb\_Sydney$host\_since, tryFormats = c("%m/%d/%y")),   
 col = brCols, pch = 18, cex = 0.5, main = "Listing Price by Host Since Year",  
 xlab = "Host Since Year", ylab = "Price per Night")  
legend(legend = c("Less than 6 beds","6-12 Beds", "12-17 Beds","17-23 Beds","23-29 Beds"), fill = rCols, "topleft")



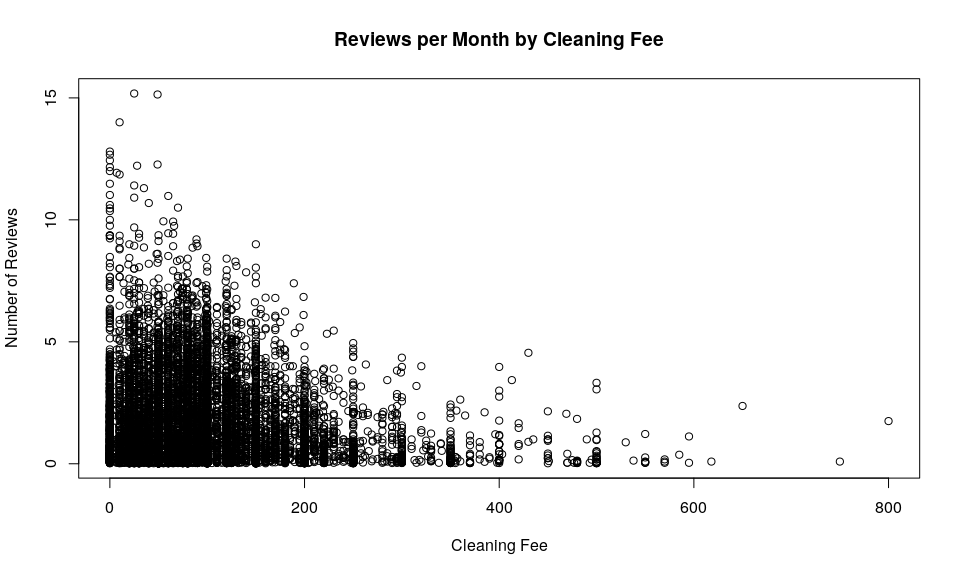
To explore my third hypothesis, I used scatterplots and a boxplot to explore how cleaning fees are related the number of bookings and review ratings for a listing. I hypothesized that listings with higher cleaning fees will have worse ratings.

The first and visual shows us that number of reviews by cleaning fee, and see can see the data is very skewed, with more reviews per month with less cleaning fees. There is a similar trend with the total number of reviews and cleaning fee. There is a steep drop in the number of reviews after 100 to 150 dollars. To look at this from another angle, I plotted property type by cleaning fee, and there are three property types that stand out in terms of cleaning fees- house, villa and apartment.

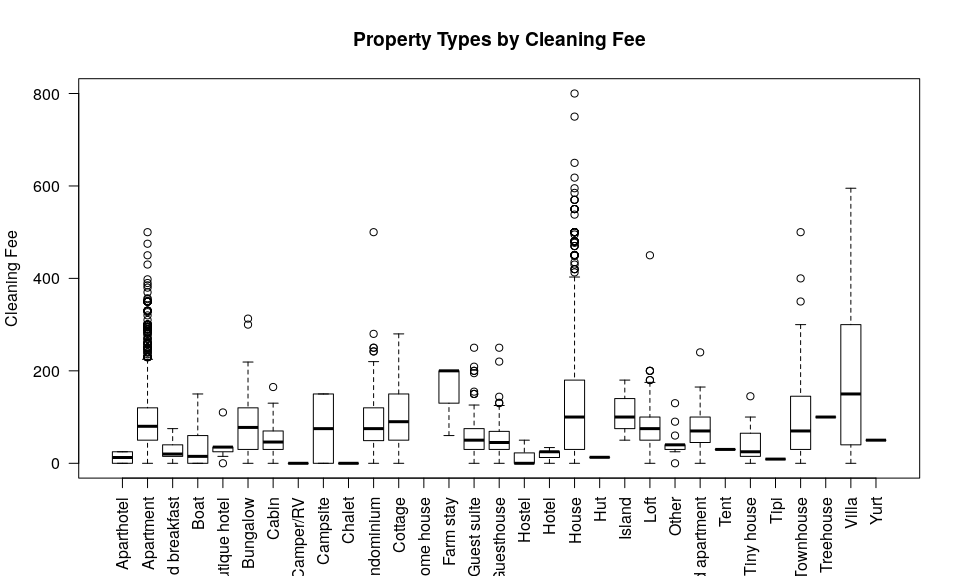
clean\_num = as.numeric(gsub("[\\$]", "", Airbnb\_Sydney$cleaning\_fee), length(2))   
Airbnb\_Sydney$clean\_num = clean\_num  
  
plot(Airbnb\_Sydney$number\_of\_reviews~clean\_num, main = "Number of Reviews by Cleaning Fee",   
 xlab = "Cleaning Fee", ylab = "Number of Reviews")



plot(Airbnb\_Sydney$reviews\_per\_month~clean\_num, main = "Reviews per Month by Cleaning Fee",   
 xlab = "Cleaning Fee", ylab = "Number of Reviews")



plot(clean\_num~as.factor(Airbnb\_Sydney$property\_type), las=2,   
 main = "Property Types by Cleaning Fee",   
 xlab = "", ylab = "Cleaning Fee")



#### Q7

##### 7.1

I computed this step eariler. Here is the code again:

price\_num = as.numeric(gsub("[\\$]", "", Airbnb\_Sydney$price), length(2))

## Warning: NAs introduced by coercion

head(price\_num)

## [1] 100 471 109 450 159 84

##### 7.2

# convert to character  
am\_char = as.character.factor(Airbnb\_Sydney$amenities)  
  
# split on comma  
am\_char\_split = strsplit(am\_char,',')  
  
# iterate over and get length for each item in list  
am\_length = lapply(am\_char\_split, length)  
head(am\_length)

## [[1]]  
## [1] 29  
##   
## [[2]]  
## [1] 20  
##   
## [[3]]  
## [1] 43  
##   
## [[4]]  
## [1] 23  
##   
## [[5]]  
## [1] 23  
##   
## [[6]]  
## [1] 21

# add to dataframe  
Airbnb\_Sydney$am\_length = I(am\_length)

##### 7.3

Below, I looked the average review score rating for each cancellation policy. The data is very skewed, with only a few values for the “strict 30 days” and “strict 60 days” categoies. We can see the ratings mean for the “strict 30 days” is the lowest at 80. The moderate, flexible and 14 day cancellation policy all have high review ratings between 93-95. This is further illustrated in the visual.

rev\_cancel = Airbnb\_Sydney[,c("review\_scores\_rating", "cancellation\_policy", "host\_is\_superhost")]  
  
moderate = rev\_cancel[rev\_cancel$cancellation\_policy=="moderate",]  
flexible = rev\_cancel[rev\_cancel$cancellation\_policy=="flexible",]  
fourteen = rev\_cancel[rev\_cancel$cancellation\_policy=="strict\_14\_with\_grace\_period",]  
thirty = rev\_cancel[rev\_cancel$cancellation\_policy=="super\_strict\_30",]  
sixty = rev\_cancel[rev\_cancel$cancellation\_policy=="super\_strict\_60",]  
  
mod\_mean = mean(moderate$review\_scores\_rating)  
flex\_mean = mean(flexible$review\_scores\_rating)  
fourteen\_mean = mean(fourteen$review\_scores\_rating, na.rm = T)  
thirty\_mean = mean(thirty$review\_scores\_rating)  
sixty\_mean = mean(sixty$review\_scores\_rating)  
  
mod\_mean

## [1] 95.00604

nrow(moderate)

## [1] 3314

flex\_mean

## [1] 94.15888

nrow(flexible)

## [1] 1391

fourteen\_mean

## [1] 93.77102

nrow(fourteen)

## [1] 6089

thirty\_mean

## [1] 80

nrow(thirty)

## [1] 1

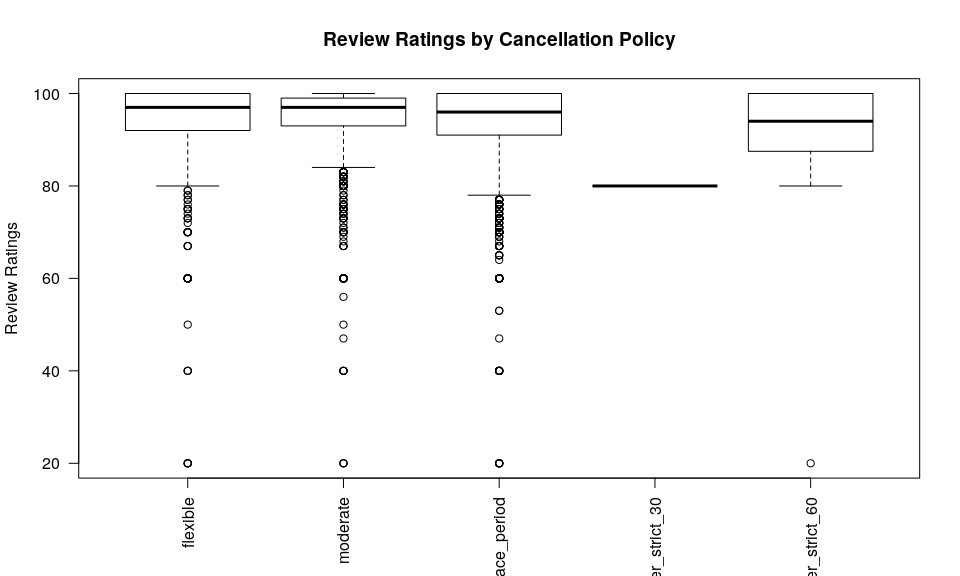
sixty\_mean

## [1] 89.8

nrow(sixty)

## [1] 20

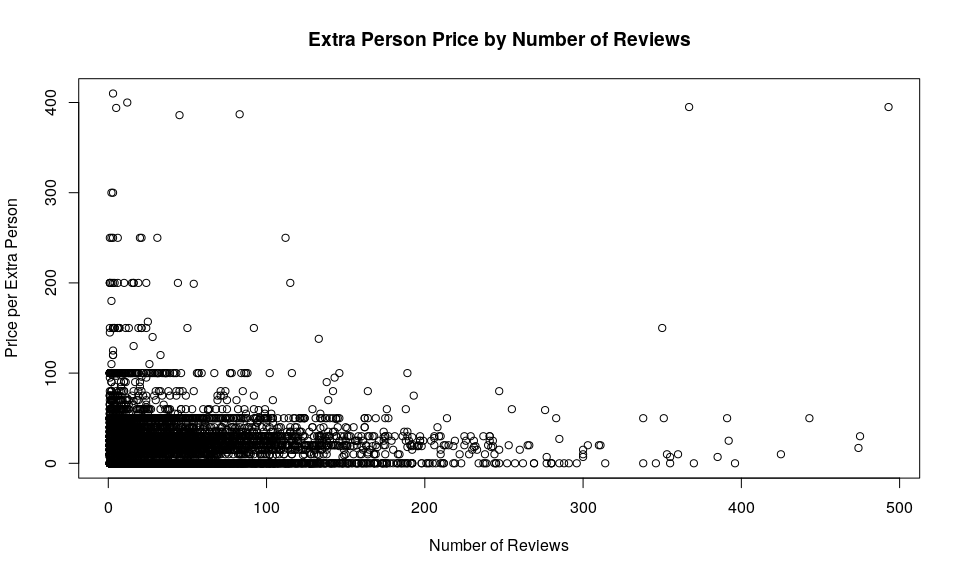
plot(rev\_cancel$review\_scores\_rating~rev\_cancel$cancellation\_policy, las=2, xlab="", ylab = "Review Ratings",  
 main = "Review Ratings by Cancellation Policy")



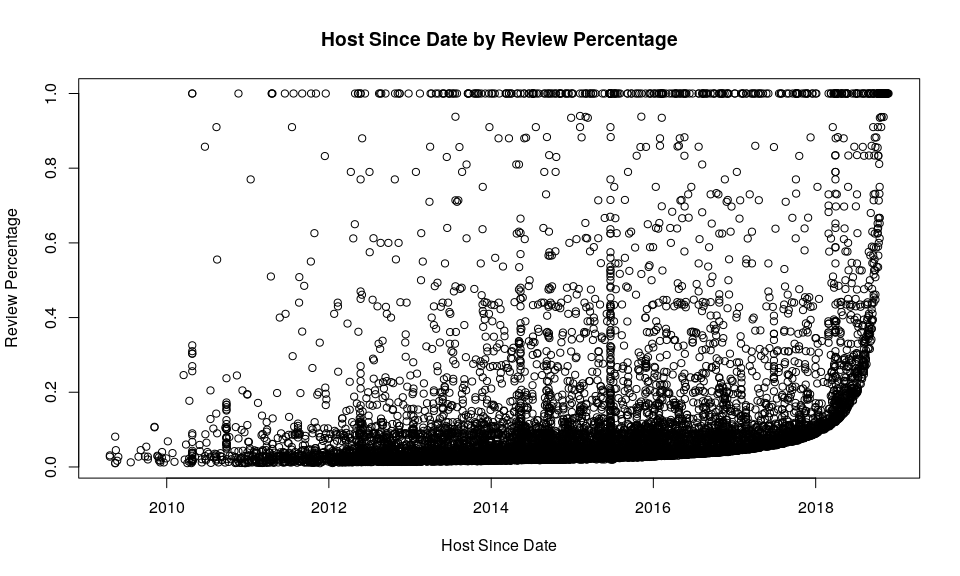
##### 7.4

Below, I cleaned and plotted the extra people variable, which is the price to have one extra people stay one night at a listing. I looked at this variable in comparison to reviews per month divided by total number of reviews. I looked at a number of variables I cleaned to compare including cancellation polices and super hosts, host since data and reviews per month divided by total number of reviews, and extra person price by reviews per month divided by total number of reviews.

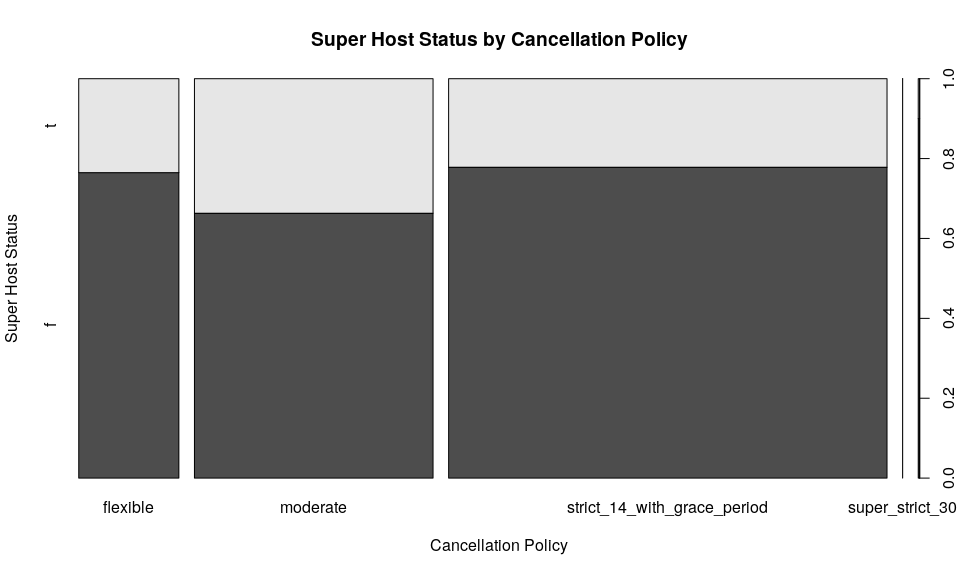
# clean extra person price and add to dataset  
extra\_num = as.numeric(gsub("[\\$]", "", Airbnb\_Sydney$extra\_people), length(2))  
Airbnb\_Sydney$extra\_num = extra\_num  
  
# add reviews per month divided by total number of reviews to dataset  
Airbnb\_Sydney$review\_perc = Airbnb\_Sydney$reviews\_per\_month/Airbnb\_Sydney$number\_of\_reviews  
  
plot(Airbnb\_Sydney$extra\_num~Airbnb\_Sydney$number\_of\_reviews, main = "Extra Person Price by Number of Reviews",  
 ylab = "Price per Extra Person", xlab = "Number of Reviews")



plot(Airbnb\_Sydney$review\_perc~Airbnb\_Sydney$host\_data, main = "Host Since Date by Review Percentage",  
 ylab = "Review Percentage", xlab = "Host Since Date")



plot(rev\_cancel$host\_is\_superhost~rev\_cancel$cancellation\_policy, main = "Super Host Status by Cancellation Policy",  
 ylab = "Super Host Status", xlab = "Cancellation Policy")



#### Q8

I chose reviews per month as my primary variable, which I think will offer more insights into possible buisness actions I can recommend. Reviews per month offers a snapshot of a listing’s activity, and we can look at other variables in more detail with this as the predictor.

I examined price, number of amenities, number of host verifications, review ratings, the price of extra people, number of people a listing accommodates, cancellation policies, number of bathrooms, number of bedrooms, and the number of beds as possible response variables. I did find a possible positive relationship between the number of amenities and reviews per month, and a possible positive relationship between the number of people accommodated and reviews per month, in addition to the relationship I outline below.

Below, I’ve created a model to investigate a relationship between the reviews per month and the cleaning fees. The linear model shows that the residuals are not evenly distributed around the mean (or symmetrical), so the model points are far away from the actual points in some areas. So the model isn’t an ideal fit. This could be an area for further investigation.

The model also shows that for the average number of reviews per month, there is a cleaning fee of roughly 103 dollars, and when the number of reviews per month increases by 1, the cleaning fee drops by roughly 5 dollars. So there is a possible negative relationship between cleaning fees and reviews per month.

We can see from the scatter plot and abline that there is a possible negative relationship. It’s not an entirely linear relationship, as we can see there is a steep curve downward between 0-2 number of reviews. It appears that listings with high cleaning fees don’t have a lot of reviews per month. This could be due to a number of reasons. High cleaning fees could belong to the more expensive properities which are rented less in general, or could be in locations that have less rentals.

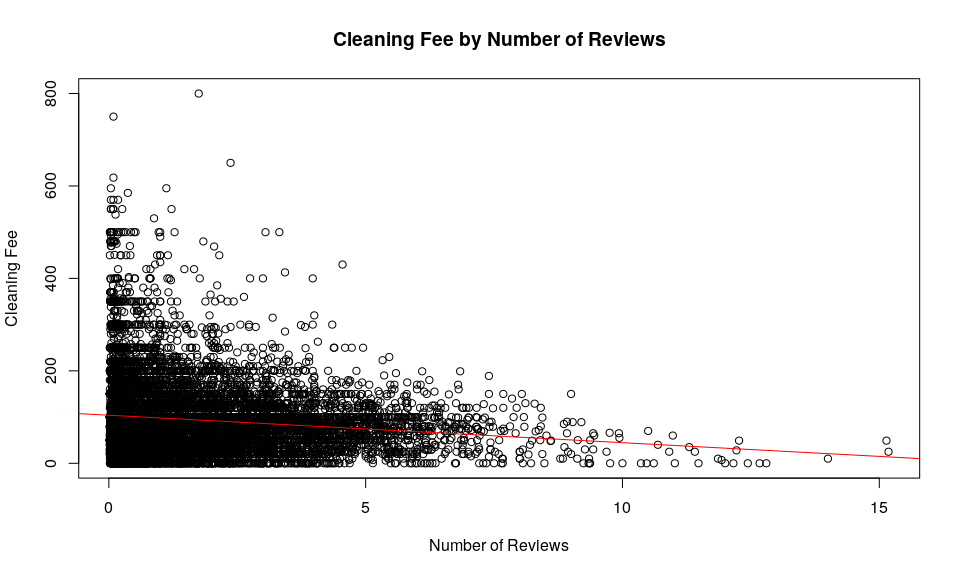
According to the model, there is a statistically significant relationship here, as the p values are close to zero and below 5%.

clean\_model = summary(lm(clean\_num~Airbnb\_Sydney$reviews\_per\_month))  
clean\_model

##   
## Call:  
## lm(formula = clean\_num ~ Airbnb\_Sydney$reviews\_per\_month)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -103.80 -53.51 -14.80 33.83 706.48   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 103.9219 1.0595 98.09 <2e-16 \*\*\*  
## Airbnb\_Sydney$reviews\_per\_month -5.9414 0.4476 -13.27 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 79.21 on 10192 degrees of freedom  
## (621 observations deleted due to missingness)  
## Multiple R-squared: 0.01699, Adjusted R-squared: 0.01689   
## F-statistic: 176.2 on 1 and 10192 DF, p-value: < 2.2e-16

plot(clean\_num~Airbnb\_Sydney$reviews\_per\_month, main = "Cleaning Fee by Number of Reviews", ylab = "Cleaning Fee", xlab = "Number of Reviews")  
abline(clean\_model, col = "Red")

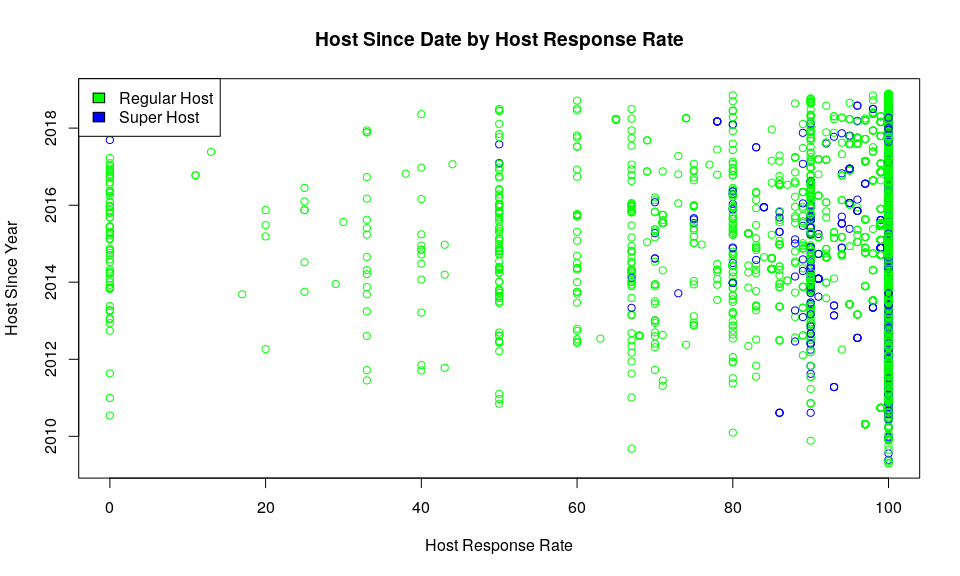
## Warning in abline(clean\_model, col = "Red"): only using the first two of 8  
## regression coefficients



## Part 3

#### Q 9.1

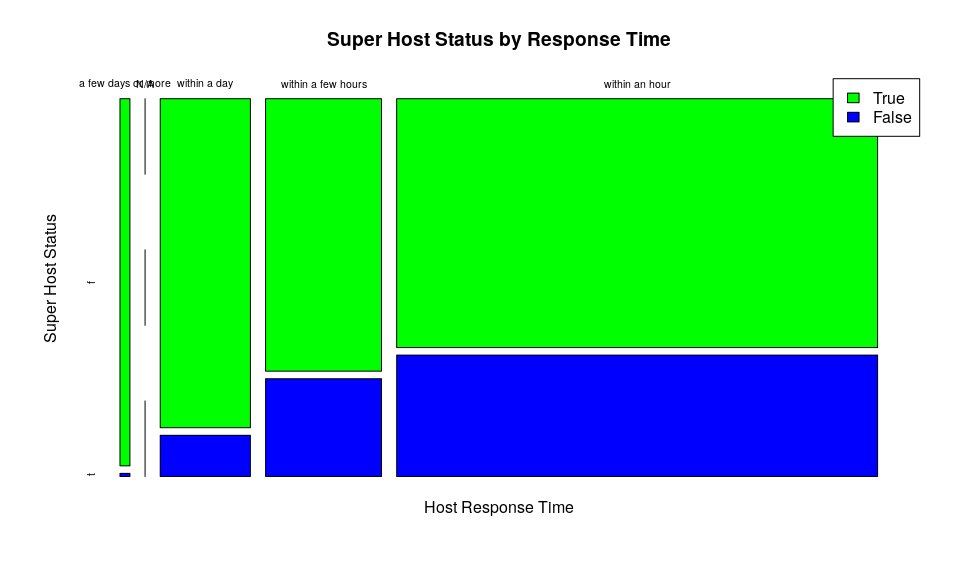
host\_response\_num = as.numeric(gsub("%", "", Airbnb\_Sydney$host\_response\_rate), length(2))  
  
# look at host response rate by host since date, and super host status  
colors = c("Green", "Blue")  
  
plot(Airbnb\_Sydney$host\_data~host\_response\_num, col = colors[Airbnb\_Sydney$host\_is\_superhost],  
 main = "Host Since Date by Host Response Rate", ylab = "Host Since Year", xlab = "Host Response Rate")  
legend(legend = c("Regular Host", "Super Host"), fill= colors, "topleft")



##### 9.2

From the mosaic plot below, we can see that super hosts tend to respond faster than regular hosts.

mosaicplot(table(Airbnb\_Sydney$host\_response\_time,Airbnb\_Sydney$host\_is\_superhost),   
 color = c("Green", "Blue"), main = "Super Host Status by Response Time",  
 ylab = "Super Host Status", xlab = "Host Response Time")  
legend(legend = c("True", "False"), fill = c("Green", "Blue"), "topright")



#### Q10

##### 10.1

The top ten words suggest that aspects of the property are mentioned often, such as the type of property, rooms within the property, and the location of the property. Also proximity words such as ‘walk’ appear, also an indicator for importance of location.

# clean and split up description data  
des = strsplit(gsub("[^[:alnum:] ]", "", Airbnb\_Sydney$description), " +")  
des\_unique = length(unique(unlist(des)))  
  
# make everything lower  
des\_lower = tolower(unlist(des))  
  
# put into table and dataframe  
des\_freq = as.data.frame(table(des\_lower))  
  
stop\_words = c("a", "able", "about", "across", "after", "all", "almost", "also", "am", "among", "an", "and", "any", "are", "as",  
"at", "be", "because", "been", "but", "by", "can", "cannot", "could", "dear", "did", "do", "does", "either", "else", "ever", "every", "for",  
"from", "get", "got", "had", "has", "have", "he", "her", "hers", "him", "his", "how", "however", "i", "if", "in", "into", "is", "it", "its", "just",  
"least", "let", "like", "likely", "may", "me", "might", "most", "must", "my", "neither", "no", "nor", "not", "of", "off", "often", "on",  
"only", "or", "other", "our", "own", "rather", "said", "say", "says", "she", "should", "since", "so", "some", "than", "that", "the", "their",  
"them", "then", "there", "these", "they", "this", "is", "to", "too", "was", "us", "wants", "was", "we", "were", "what", "when", "where",  
"which", "while", "who", "whom", "why", "will", "with", "would", "yet", "you", "your")  
  
# remove stop words  
top\_des = as.data.frame(des\_freq[!(des\_freq$des\_lower%in%stop\_words),])  
  
head(top\_des[order(top\_des$Freq, decreasing = T),], n=10L)

## des\_lower Freq  
## 2588 apartment 14622  
## 23835 walk 10469  
## 3851 bedroom 10176  
## 21810 sydney 9657  
## 19034 room 9544  
## 13047 kitchen 9072  
## 3616 beach 8855  
## 3778 bed 8702  
## 11756 house 7151  
## 569 2 7137

##### 10.2

The averages below show that lsitings with descriptions with the word ‘beach’ or ‘beaches’ in them are priced moderately higher than listings without those words in the description.

Airbnb\_Sydney$price\_num = price\_num  
# find words by subsetting and grep()  
the\_beach = Airbnb\_Sydney[grep("beach", Airbnb\_Sydney$description), c("host\_id","price\_num")]  
the\_beaches = Airbnb\_Sydney[grep("beaches", Airbnb\_Sydney$description), c("host\_id","price\_num")]  
  
# means  
mean(the\_beach$price\_num, na.rm = T)

## [1] 218.986

mean(the\_beaches$price\_num, na.rm=T)

## [1] 216.5746

mean(Airbnb\_Sydney$price\_num, na.rm = T)

## [1] 189.3062

top\_freq = function(input, word){  
 # Find high frequency words in a dataset. Input is a vector and the string to look for. Will return a row from a tabel with the word and the frequency.   
 des = strsplit(gsub("[^[:alnum:] ]", "", input), " +")  
 des\_lower = tolower(unlist(des))  
 des\_freq = as.data.frame(table(des\_lower))  
 top\_des = as.data.frame(des\_freq[!(des\_freq$des\_lower%in%stop\_words),])  
 word\_freq = top\_des[top\_des$des\_lower==word,]  
 returnValue(word\_freq)  
}  
  
# calling function  
  
top\_freq(Airbnb\_Sydney$description, "apartment")

## des\_lower Freq  
## 2588 apartment 14622

top\_freq(Airbnb\_Sydney$description, "bed")

## des\_lower Freq  
## 3778 bed 8702

row\_freq = function(input, word){  
 # function to indicate the frequency of a word in a list. Input is a list of vectors and the string to look for. Will return a list with integers to indicate the frequency of a word. ) means that word does not appear in that row.   
 des = strsplit(gsub("[^[:alnum:] ]", "", input), " +")  
 pattern = lapply(des, grep, pattern=word)  
 lens = lapply(pattern, length)  
 returnValue(lens)  
}  
  
head(row\_freq(Airbnb\_Sydney$description, "beach"))

## [[1]]  
## [1] 0  
##   
## [[2]]  
## [1] 3  
##   
## [[3]]  
## [1] 0  
##   
## [[4]]  
## [1] 0  
##   
## [[5]]  
## [1] 0  
##   
## [[6]]  
## [1] 1

##### 10.3

I looked the words ‘apartment’, ‘bed’, and ‘house’. I can see the average price of listings with those words differs quite a bit. I also compared this with super host status for further insight.

the\_apt = Airbnb\_Sydney[grep("apartment", Airbnb\_Sydney$description),  
 c("host\_is\_superhost", "price\_num")]  
the\_bed = Airbnb\_Sydney[grep("bed", Airbnb\_Sydney$description),   
 c("price\_num", "host\_is\_superhost")]  
sup\_house = Airbnb\_Sydney[grep("house", Airbnb\_Sydney$description),   
 c("host\_is\_superhost", "price\_num")]  
  
mean(the\_apt$price\_num, na.rm = T)

## [1] 171.7812

mean(the\_bed$price\_num, na.rm = T)

## [1] 190.3527

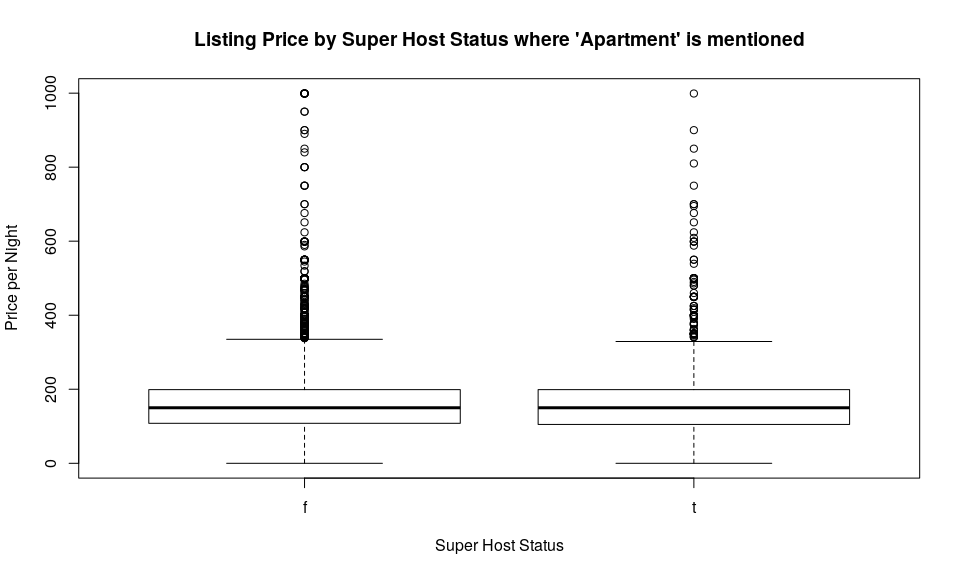
mean(sup\_house$price\_num, na.rm = T)

## [1] 209.9777

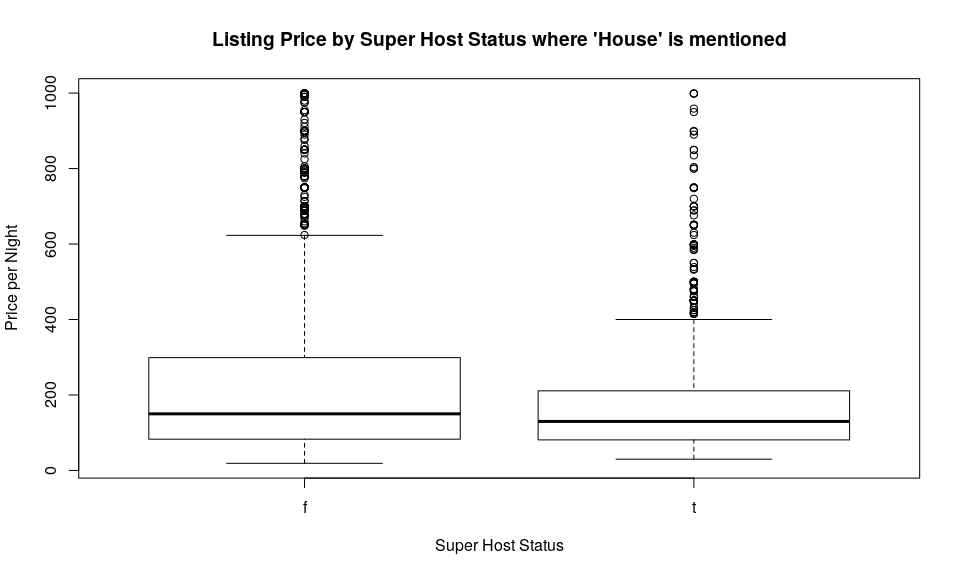
mean(Airbnb\_Sydney$price\_num, na.rm = T)

## [1] 189.3062

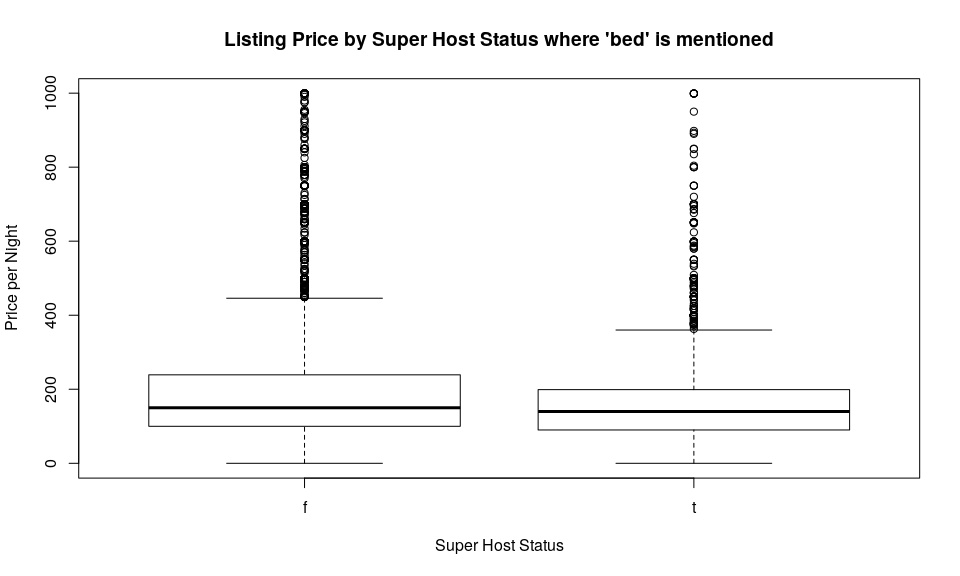
plot(the\_apt, main = "Listing Price by Super Host Status where 'Apartment' is mentioned", xlab = "Super Host Status", ylab = "Price per Night")



plot(sup\_house, main = "Listing Price by Super Host Status where 'House' is mentioned", xlab = "Super Host Status", ylab = "Price per Night")



plot(as.numeric(the\_bed$price\_num)~the\_bed$host\_is\_superhost, main = "Listing Price by Super Host Status where 'bed' is mentioned", xlab = "Super Host Status", ylab = "Price per Night")



###### 10-2

###### 10-2.1

I chose to look at cities, because I thought zipcodes would include a wider area, which could introduce more variability in the data. Cities seemed liked a good way to hold variables like price constant.

# Use for loop to get the number of host ids for each unique city name and add to a dataframe  
cities = data.frame()  
for(i in unique(Airbnb\_Sydney$city)){  
 city\_test = Airbnb\_Sydney[Airbnb\_Sydney$city==i, c("host\_id")]  
 city\_len = length(Airbnb\_Sydney[Airbnb\_Sydney$city==i, c("host\_id")])  
 cities1 = data.frame("City" = i,city\_len)  
 cities = rbind(cities, cities1)  
 returnValue(cities)  
}  
  
# print the top 100 cities with the most listings  
head(cities[order(cities$city\_len, decreasing = T),], n=100L)

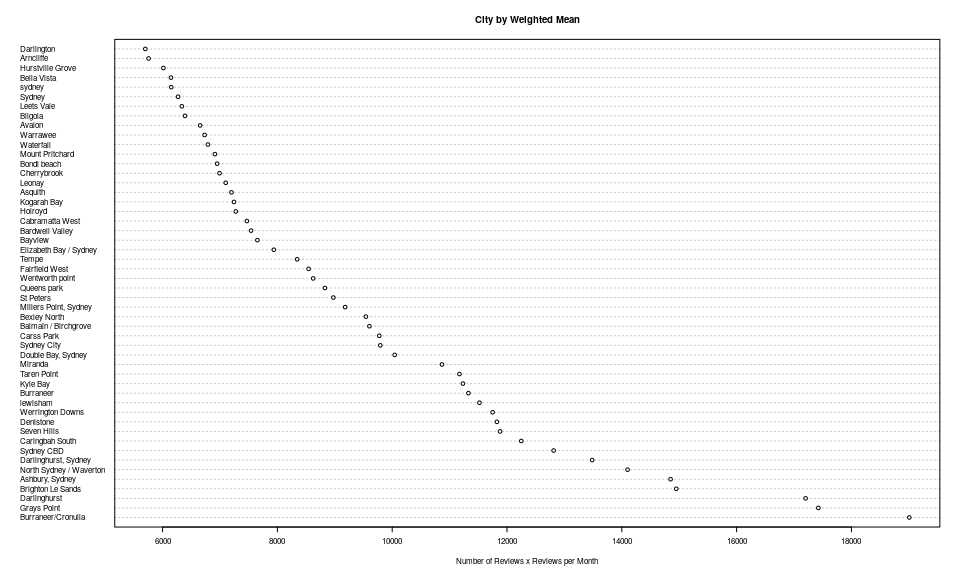
## City city\_len  
## 7 Bondi Beach 555  
## 49 Surry Hills 500  
## 11 Sydney 463  
## 73 Manly 389  
## 3 Darlinghurst 373  
## 16 Coogee 272  
## 21 Bondi 222  
## 79 Randwick 221  
## 51 Redfern 206  
## 27 Potts Point 198  
## 6 North Bondi 191  
## 39 Newtown 190  
## 1 Pyrmont 178  
## 60 Paddington 175  
## 8 Mosman 167  
## 48 Chippendale 155  
## 19 Bronte 149  
## 42 Bondi Junction 142  
## 68 Maroubra 139  
## 36 Waterloo 131  
## 107 Avalon Beach 118  
## 32 Ultimo 115  
## 13 Elizabeth Bay 101  
## 116 Mascot 95  
## 44 Glebe 93  
## 86 Zetland 93  
## 184 Haymarket 92  
## 46 Marrickville 90  
## 188 Sydney Olympic Park 89  
## 37 Camperdown 88  
## 23 Erskineville 87  
## 9 Alexandria 83  
## 63 Fairlight 83  
## 4 Balmain 79  
## 93 Leichhardt 76  
## 83 Annandale 75  
## 62 Clovelly 72  
## 66 Tamarama 72  
## 75 Rose Bay 69  
## 41 Bellevue Hill 67  
## 40 Rozelle 66  
## 5 North Sydney 65  
## 90 Rushcutters Bay 64  
## 29 Woollahra 63  
## 18 Neutral Bay 62  
## 56 Woolloomooloo 60  
## 64 Rosebery 59  
## 119 Freshwater 59  
## 71 Palm Beach 57  
## 2 Balgowlah 56  
## 85 Waverley 52  
## 112 Cronulla 50  
## 121 Forest Lodge 50  
## 251 Chatswood 50  
## 22 Newport 49  
## 31 Kirribilli 49  
## 50 Cremorne 48  
## 77 Vaucluse 46  
## 137 Dee Why 46  
## 179 Ashfield 46  
## 30 Double Bay 43  
## 92 Arncliffe 43  
## 316 Wolli Creek 43  
## 84 Millers Point 39  
## 204 Rhodes 36  
## 69 Kensington 35  
## 94 Queenscliff 35  
## 142 Strathfield 35  
## 143 Burwood 35  
## 148 Kingsford 35  
## 105 Birchgrove 34  
## 122 Dulwich Hill 34  
## 53 Stanmore 33  
## 70 Darlington 32  
## 91 Lilyfield 32  
## 34 Darling Point 31  
## 76 Crows Nest 31  
## 136 Mona Vale 30  
## 320 Parramatta 30  
## 178 Cammeray 28  
## 172 Brighton-Le-Sands 27  
## 123 Manly Vale 26  
## 129 Wollstonecraft 26  
## 113 Bundeena 25  
## 132 Edgecliff 25  
## 195 North Balgowlah 25  
## 196 Ryde 25  
## 57 Lane Cove North 24  
## 89 Enmore 24  
## 104 Narrabeen 24  
## 158 McMahons Point 24  
## 96 Petersham 23  
## 45 Drummoyne 22  
## 166 Lewisham 22  
## 321 Wentworth Point 22  
## 378 Bankstown 22  
## 10 Avalon 21  
## 97 Queens Park 21  
## 111 Roseville 20  
## 98 Sans Souci 19

# calculate weighted means by multiplying the number of reviews by review ratings  
weighted = Airbnb\_Sydney$number\_of\_reviews\*Airbnb\_Sydney$review\_scores\_rating  
  
# add to dataframe  
Airbnb\_Sydney$weigted = weighted  
  
# subset for certain columns  
weight = Airbnb\_Sydney[,c("host\_id","weigted","city")]  
  
# aggregate data by city and get mean, print top 100  
head(aggregate(weight[2], by=list(weight$city), mean), n=100L)

## Group.1 weigted  
## 1 3463.7500  
## 2 • Darling harbour 637.0000  
## 3 Abbotsford 1201.8750  
## 4 Agnes Banks 1500.0000  
## 5 Alexandria 2340.7470  
## 6 Alexandria 3648.0000  
## 7 Allambie Heights 5078.9091  
## 8 Allawah 1020.0000  
## 9 Allawah/Carlton 3267.0000  
## 10 Annandale 2520.6533  
## 11 Arcadia 843.0000  
## 12 Arncliffe 5755.6279  
## 13 Artarmon 1329.7143  
## 14 Ashbury, Sydney 14850.0000  
## 15 Ashfield 1690.9565  
## 16 Ashfield, New South Wales, AU 651.0000  
## 17 Asquith 7200.0000  
## 18 Auburn 3999.0000  
## 19 Auburn 1692.0000  
## 20 Auburn / Lidcomb 582.0000  
## 21 Avalon 6654.8095  
## 22 Avalon Beach 3073.3644  
## 23 Balgowlah 1228.1786  
## 24 Balgowlah Heights 1237.8333  
## 25 Balmain 2605.7595  
## 26 Balmain / Birchgrove 9603.0000  
## 27 Balmain East 2640.1667  
## 28 Balmoral Beach 422.0000  
## 29 Bangor 2570.5000  
## 30 Banksia 682.6667  
## 31 Banksia Sydney 200.0000  
## 32 Bankstown 1356.9545  
## 33 Bar Point 1000.0000  
## 34 Barangaroo 2466.5000  
## 35 Bardia 810.0000  
## 36 Bardwell Valley 7541.3333  
## 37 Barpoint 300.0000  
## 38 Baulkham Hills 2258.2857  
## 39 Bayview 7651.3333  
## 40 Beacon Hill 1978.7857  
## 41 Beaconsfield 3864.0769  
## 42 Beaumont Hills 2718.0000  
## 43 Beecroft 1255.5000  
## 44 Belfield 692.0000  
## 45 Bella Vista 6145.7143  
## 46 Bellevue Hill 1484.3433  
## 47 Bellevue Hill (Double Bay side). 1372.0000  
## 48 Bellevue Hill, Sydney 200.0000  
## 49 Belmore 861.7500  
## 50 Berala 742.6000  
## 51 Berowra Creek 300.0000  
## 52 Berowra Heights 2857.0000  
## 53 Berowra Waters 1504.0000  
## 54 Beverly Hills 3433.2000  
## 55 Bexley 3461.5000  
## 56 Bexley North 9541.6667  
## 57 Bilgola 6390.5000  
## 58 Bilgola Beach 2243.5294  
## 59 Bilgola Plateau 1528.9375  
## 60 Bilgola, Sydney 375.0000  
## 61 Birchgrove 1556.2059  
## 62 Blacktown 613.5000  
## 63 Blair Athol 100.0000  
## 64 Blakehurst 919.0000  
## 65 Bondi 1960.8198  
## 66 Bondi 341.0000  
## 67 bondi beach 1700.0000  
## 68 Bondi Beach 2376.6883  
## 69 Bondi beach 6950.0000  
## 70 Bondi Beach, Sydney 639.0000  
## 71 Bondi Junction 2065.4507  
## 72 Bondi Junction 80.0000  
## 73 Bondi Junction Sydney 3977.0000  
## 74 Bondi Junction, Sydney 3284.0000  
## 75 Bondi, Tamarama 712.0000  
## 76 Botany 1529.3571  
## 77 Breakfast Point 784.0000  
## 78 Brighton Le Sands 14949.0000  
## 79 Brighton-Le-Sands 1392.2963  
## 80 Bronte NA  
## 81 Bronte 300.0000  
## 82 Brooklyn 4985.0000  
## 83 Brookvale 460.0000  
## 84 Bundeena 3212.0400  
## 85 Bungarribee 1140.0000  
## 86 Burraneer 11328.0000  
## 87 Burraneer/Cronulla 19008.0000  
## 88 Burwood 1461.1143  
## 89 Cabarita 180.0000  
## 90 Cabramatta 1343.6667  
## 91 Cabramatta West 7469.0000  
## 92 Cammeray 2864.9643  
## 93 Campbelltown 869.0000  
## 94 camperdown 1316.0000  
## 95 Camperdown 1854.8068  
## 96 Campsie 1570.6000  
## 97 Canada Bay 100.0000  
## 98 Canley Heights 973.5000  
## 99 Canterbury 4982.0000  
## 100 Caringbah 1045.0000

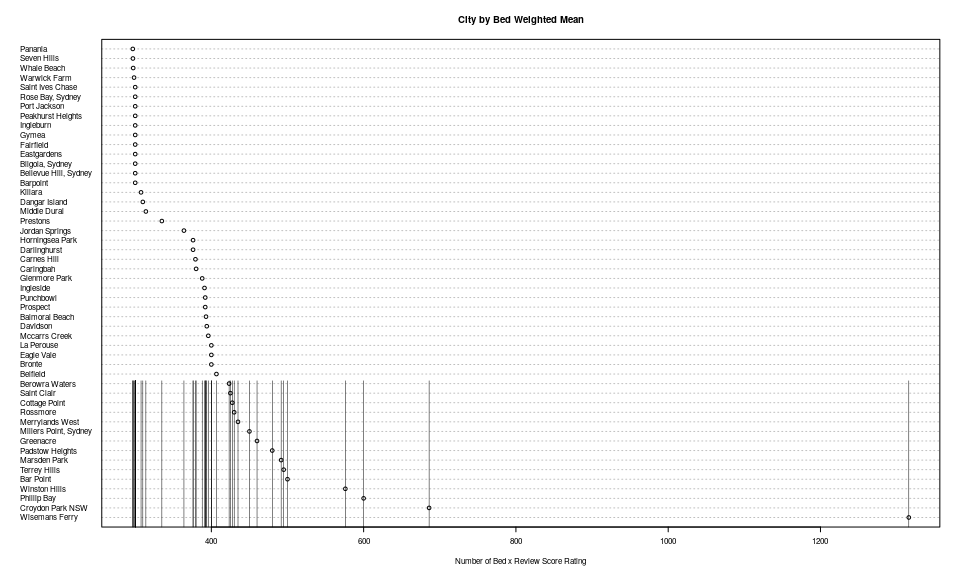
From this graph we can see a clear downward trend when the cities are organized by weighted mean. We can see which cities have a high number of total reviews and reviews per month, and what that average is. This tells us which cities have the most sustained, overall activity.

# subet for a separate dataframe  
city\_df = Airbnb\_Sydney[,c("city","number\_of\_reviews","review\_scores\_rating")]  
  
# calculate weighted means  
m\_w = city\_df$number\_of\_reviews \* city\_df$review\_scores\_rating  
city\_df$weights = m\_w  
  
# use aggregate function  
city\_agg = aggregate(city\_df$weights,   
 by=list(city = city\_df$city),   
 mean)  
  
city\_100 = head(city\_agg[order(city\_agg$x, decreasing = T),], n=50L)  
  
dotchart(city\_100$x, labels = city\_100$city, cex = .5, main = "City by Weighted Mean", xlab = "Number of Reviews x Reviews per Month")

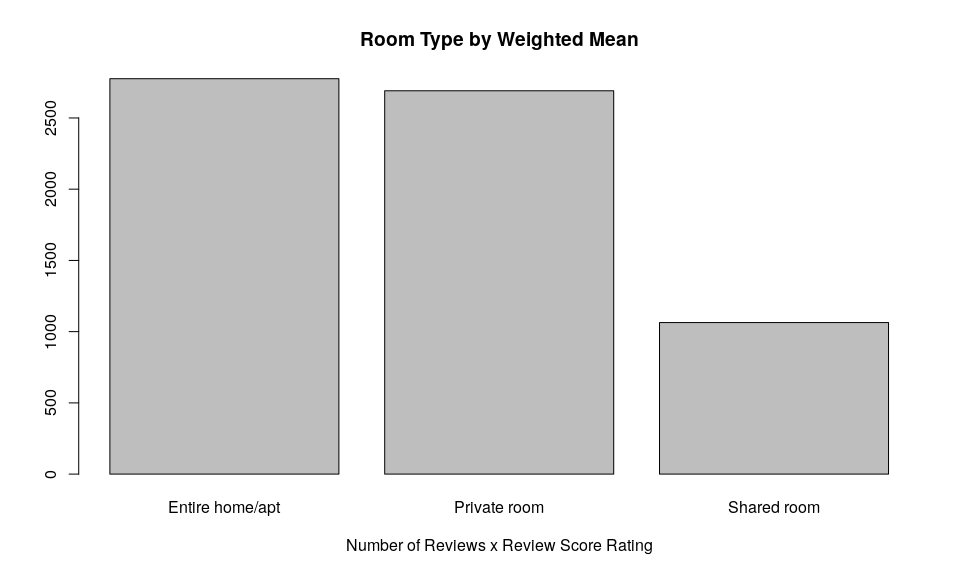


##### 10-2.2

city\_beds = Airbnb\_Sydney[,c("city","bedrooms","review\_scores\_rating")]  
  
  
m\_b = city\_beds$bedrooms \* city\_beds$review\_scores\_rating  
city\_beds$weights = m\_b  
  
city\_beds\_agg = aggregate(city\_beds$weights,   
 by=list(city = city\_beds$city),   
 mean)  
  
beds\_100 = head(city\_beds\_agg[order(city\_beds\_agg$x, decreasing = T),], n=50L)  
  
dotchart(beds\_100$x, labels = beds\_100$city, cex = .5, main = "City by Bed Weighted Mean", xlab = "Number of Bed x Review Score Rating")  
  
rug(beds\_100$x, ticksize = 0.3)



rooms = Airbnb\_Sydney[,c("room\_type","number\_of\_reviews","review\_scores\_rating")]  
  
r\_r = rooms$number\_of\_reviews \* rooms$review\_scores\_rating  
rooms$weights = r\_r  
  
rooms\_agg = aggregate(rooms$weights,   
 by=list(room\_type = rooms$room\_type),   
 mean, na.rm =T)  
  
barplot(rooms\_agg$x, names.arg = rooms\_agg$room\_type, main = "Room Type by Weighted Mean", xlab = "Number of Reviews x Review Score Rating")



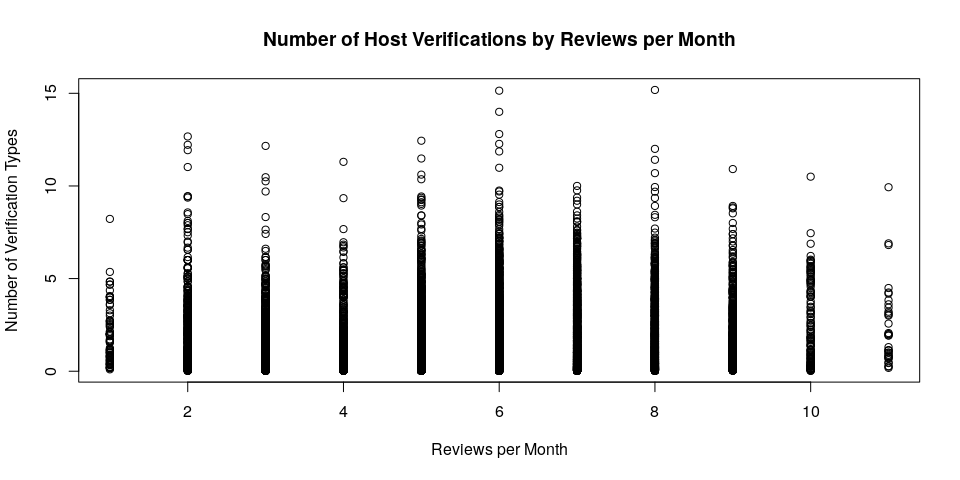
## Part 4

Below, I looked at the host verfication information for further insights. I split and cleaned verification information and plotted super host information and reviews per month. I also looked at a possbile linear relationship between the number of host verifications and reviews per month, and saw a positive relationship. We can also see some verification types are used way more than others, and that super hosts tend to have more types of verfication.

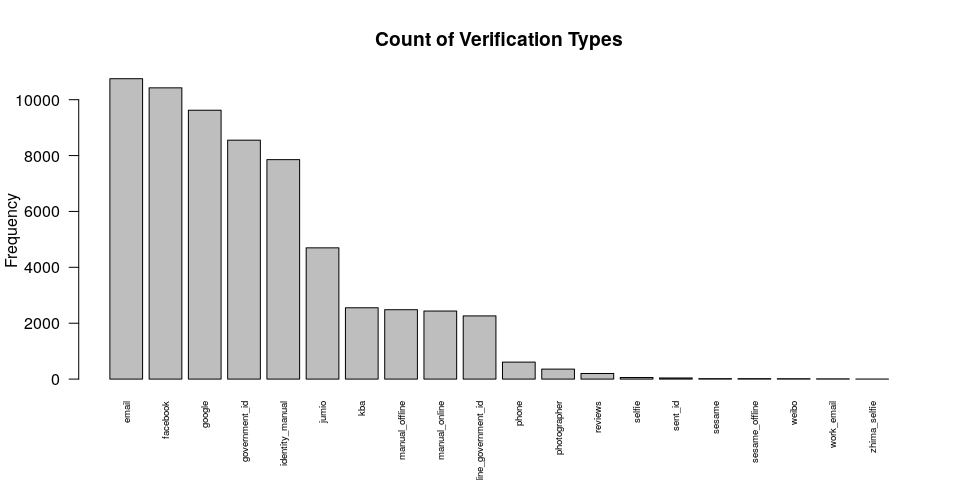
# convert and split host verification  
ver\_char = as.character.factor(Airbnb\_Sydney$host\_verifications)  
ver\_char\_split = strsplit(ver\_char,',')  
  
# get number of verfications, assign to dataframe  
ver\_length = lapply(ver\_char\_split, length)  
Airbnb\_Sydney$ver\_length = I(ver\_length)  
  
# clean data  
  
ver\_sub1 = gsub("^[\\[']", "", unlist(ver\_char\_split))  
  
ver\_sub2 = gsub("\\']", "", ver\_sub1)  
  
ver\_sub3 = gsub(" '", "", ver\_sub2)  
  
ver\_sub4 = gsub("'","", ver\_sub3)  
  
ver\_sub5 = gsub("]", NA, ver\_sub4)  
  
ver\_df = as.data.frame(table(ver\_sub5))  
  
# fit linear model  
  
summary(lm(as.numeric(Airbnb\_Sydney$ver\_length)~Airbnb\_Sydney$reviews\_per\_month))

##   
## Call:  
## lm(formula = as.numeric(Airbnb\_Sydney$ver\_length) ~ Airbnb\_Sydney$reviews\_per\_month)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.5153 -0.8509 0.2101 1.2769 5.3239   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.65729 0.02446 231.30 <2e-16 \*\*\*  
## Airbnb\_Sydney$reviews\_per\_month 0.10439 0.01042 10.02 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.89 on 10813 degrees of freedom  
## Multiple R-squared: 0.009204, Adjusted R-squared: 0.009112   
## F-statistic: 100.4 on 1 and 10813 DF, p-value: < 2.2e-16

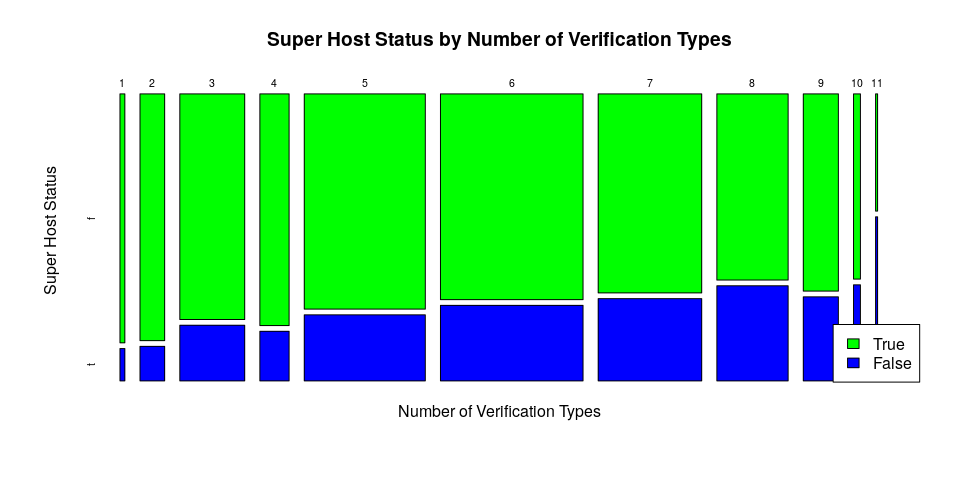
# plot linear model  
plot(Airbnb\_Sydney$reviews\_per\_month~as.numeric(Airbnb\_Sydney$ver\_length), main = "Number of Host Verifications by Reviews per Month",  
 xlab = "Reviews per Month", ylab = "Number of Verification Types")



# plot number of verification type counts  
  
barplot(ver\_df$Freq[order(ver\_df$Freq, decreasing = T)], names.arg = ver\_df$ver\_sub5, las =2,  
 main = "Count of Verification Types", ylab = "Frequency", cex.names = 0.6)



# plot number of verification types by super host status  
  
mosaicplot(table(as.numeric(Airbnb\_Sydney$ver\_length),Airbnb\_Sydney$host\_is\_superhost),   
 color = c("Green", "Blue"), main = "Super Host Status by Number of Verification Types",  
 ylab = "Super Host Status", xlab = "Number of Verification Types")  
legend(legend = c("True", "False"), fill = c("Green", "Blue"), "bottomright")



I also used a function to get the percentage of times a certain verification type was used in the dataset. We can see that percentage really high for email, phone, and facebook verification types.

ver\_f = function(text){  
 # this function take in a string of text to search verfification types and a variable name to subset with  
 # it returns the percentage of time a verification type was mentioned  
 input = grepl(text, Airbnb\_Sydney$host\_verifications)  
 yes\_input = Airbnb\_Sydney$host\_verifications[input]  
 len\_input = round(length(yes\_input)/length(Airbnb\_Sydney$host\_verifications), 3)  
 print(len\_input)  
}  
  
ver\_f("email")

## [1] 0.967

ver\_f("phone")

## [1] 0.994

ver\_f("government\_id")

## [1] 0.791

ver\_f("facebook")

## [1] 0.209

ver\_f("reviews")

## [1] 0.89

ver\_f("work\_email")

## [1] 0.225

ver\_f("jumio")

## [1] 0.726

## Part 5

My analysis focused on the verfication types, price, property types, super host, cleaning fees, review ratings, and number of reviews. I found a number of interesting patterns: 1. Super hosts tend to have more vertification types listed. They also appear to respond faster to customer inquires than regular hosts. Super hosts also had a higher response rate, with the lowest response rate at about 63 percent. So, is it worth it to be a superhost? When comparing price and superhost status, there does not seem to be a significant difference- the average and range are roughly the same. When comparing reveiw ratings and super host status, super host do have a higher average, a higher number of reviews, and more reviews per month. Also, when looking at super host status and different description words, such as ‘house’, there is a small difference in price. A business action could be taken to encourage or incentivize regular hosts to upgrade to super hosts to increase traffic and reviews. 2. Additional costs and rules such as cancellation policies and cleaning fees had an effect on the number of reviews on listing. It appears there were fewer total reviews for listings that charged extra or were stricter on policies. Even a moderate increase in cleaning fees was associated with less reviews. It could also be that listings with higher cleaning fees were also more expensive, or rented less for other reasons. However, higher charges for extra people also led to less reviews overall. A business action to address this could include encouraging hosts to roll cleaning prices into the overall price, or allowing no extra people. Any way to reduce the number of extra costs would be a good action to take.  
3. Location and property types significantly affected how often a listing is rented. Certain cities and property types where listed way more than others. Even location or property specific words in the listing description contributed to a difference in review numbers and ratings. I would recommend that hosts use certain words to attract more customers in their description. 4. Having extra perks, such as amenities, increased business. A higher number of amenities led to more reviews. I would encourage hosts to list all possible amenities in a listing to attract more buisness and reviews.

A general issue I struggled with was the skew of the data. The number of reviews and reviews per month was highly skewed, making it difficult to detect trends and find conclusions. The property type variable was also drastically skewed, with the vast majority of properities being apartments. This is reflected in my analysis and visualizations.

## Part 6

Throughout this project, I iterated over the six divisions of data science to extract as much information as possible from a secondary dataset to generate insights from the data. There were some steps that I iterated over more than others, and that includes the data exploration step. After reading the data in as a csv file, I got some basic information about the data and then started to look at individual columns. The 50 Years of Data Science article mentions that data exploration is about 80% of the work, and that felt accurate while working with this dataset. I looked at many columns individually to understand their class, missing elements, and potential for insight. To get a better understanding of the data, I created several plots to look at the distribution of individual variables, and also to take a look at descriptive statistics with different variable combinations. This helped me determine the depth and richness of the information, provided hints at what to expect in the analysis phase, and highlighted what data needed to be cleaned. Hypothesis were formed at the conclusion of the data exploration phase. The data representation and transformation process for this project was simple- we worked with a csv file and did not have to do anything complicated to read it in, or to start exploring. The computing with data step was also a non-issue with this project because we just used R, and didn’t combine any other computing methods or sources. I started the data cleaning process with some obvious fixes such as removing the dollar sign from price and converting the host\_since variable to a proper date format. I also cleaned several of the character and list variables to do text analysis. This phase also required going back and reviewing my data explortion phase to decide what columns to focus on and what classes they needed to be. Cleaning the data allowed me to aggregate and group the data to start looking for potential relationships between variables. The data analysis phase followed next to further investigate potential relationships, with more data visualization to illustrate what was found. This was part of the data visualization and presentation phase of the lifecycle. After compiling the analysis, I created visuals to show the nature of the relationships I found and to show trends in the data. This phase included raw numbers from the analysis and data visualizations. The data modeling phase was touched on briefly in this project as we did some basic statistics. This analysis leaned toward the generative analysis side of things and not so much predictive analysis. The last division, science of data science, we left out of this project. Although there is potential to look at how I understood, cleaned, analyzed and visualized this data, we did not do this here. This project required me to keep the whole project and goal in mind even as I was working on individual parts of the project. As a result, I continued to go back and work on previous steps, and slowly came to conclusions after much tweaking. I did the project in the WholeTale tool to work in a virtual environment and have a reproducable product at the end of the class. I did my best to record my data lifecycle process by documenting my findings and by adding plenty of comments in my code, to illustrate my workflow. To me, the data science lifecycle is a constant practice of the science of science, and I added to it by processing this secondary dataset for insights but also by putting forth my own workflow and methods.