Data preparation and analysis

e) Other Contributions

This R script below is used for Exploratory data analysis. Most of the plots in the preliminary analysis were implemented using ggplots. For outlier detection, we used 1.5IQR rule to reduce the skewness, after which approximately 90% of the data is retained which is a satisfactory representation. The density of the listings accross neighbourhoods were visualised using an interactive map created with the help of the 'leaflet' package. To find the most common words in reviews and listing descriptions, we used 'unnest_token' function present in the 'tidytext' package in the first step to tokenise the words from the texts. The scientific notations on the x-axis lables in the top 20 words plots were formatted using package 'scales'.

Listing price

```
#calendar has unformatted price data
calendar$price<-as.numeric(gsub('[$,]','',calendar$price))
summary(calendar$price)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   10.0    80.0   125.0   208.5   215.0 10000.0 2114655

#listings has unformatted data as well
listings$price<-as.numeric(gsub('[$,]','',listings$price))
summary(listings$price)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   10.0    80.0   119.0   206.2   199.0 13000.0

#remove na values
c<-which((is.na(calendar$price)))
calendar<-calendar[-c,] #1305030
calendar$day<-weekdays(calendar$date)
calendar$month<-month(calendar$date)
stats<- calendar %>% summarise(mean=mean(price), median=median(price),
stdDev=sd(price), q1=quantile(price,probs=0.25), q3=quantile(price,
probs=0.75),n=n())
```

mean 208.501 median 125 stdDev 340.2615 q1 80 q3 215 n 1305030

Outlier detection for listing price

```
#outlier detection for price
skewness(calendar$price) #11

[1] 11.77741

iqr<-stats$q3 - stats$q1
iqrm<-1.5*iqr
```

```
od<-stats$q3 + iqrm
ecdf(calendar$price)(od)

[1] 0.9049194

#90% of the data is retained after outlier detection

cleaned_calendar<-calendar %>%
  filter(price<od)
od

  75%
417.5

skewness(cleaned_calendar$price) #1.1

[1] 1.138918

by_month<-calendar %>%
  group_by(month) %>%
  summarise(avg_price=mean(price))

by_day<-calendar %>%
  group_by(day) %>%
  summarise(avg_price=mean(price))
```

The technique of outlier detection employed here is the 1.5IQR rule, after which approx. 90% of the data is retained which is a pretty satisfactory representation.

After cleaning the calendar for outliers, the price can be looked at from a monthly and day granularity level. The month of June saw the highest average price of around $220 while February saw the least of around $189

```
#Finding min and max values in the calendar data
min(calendar$date)

[1] "2018-11-15"

max(calendar$date)

[1] "2019-11-20"
```

Another point to note is that the price and availability data is available for approx. a year i.e. from November 15 2018 to November 20 2019

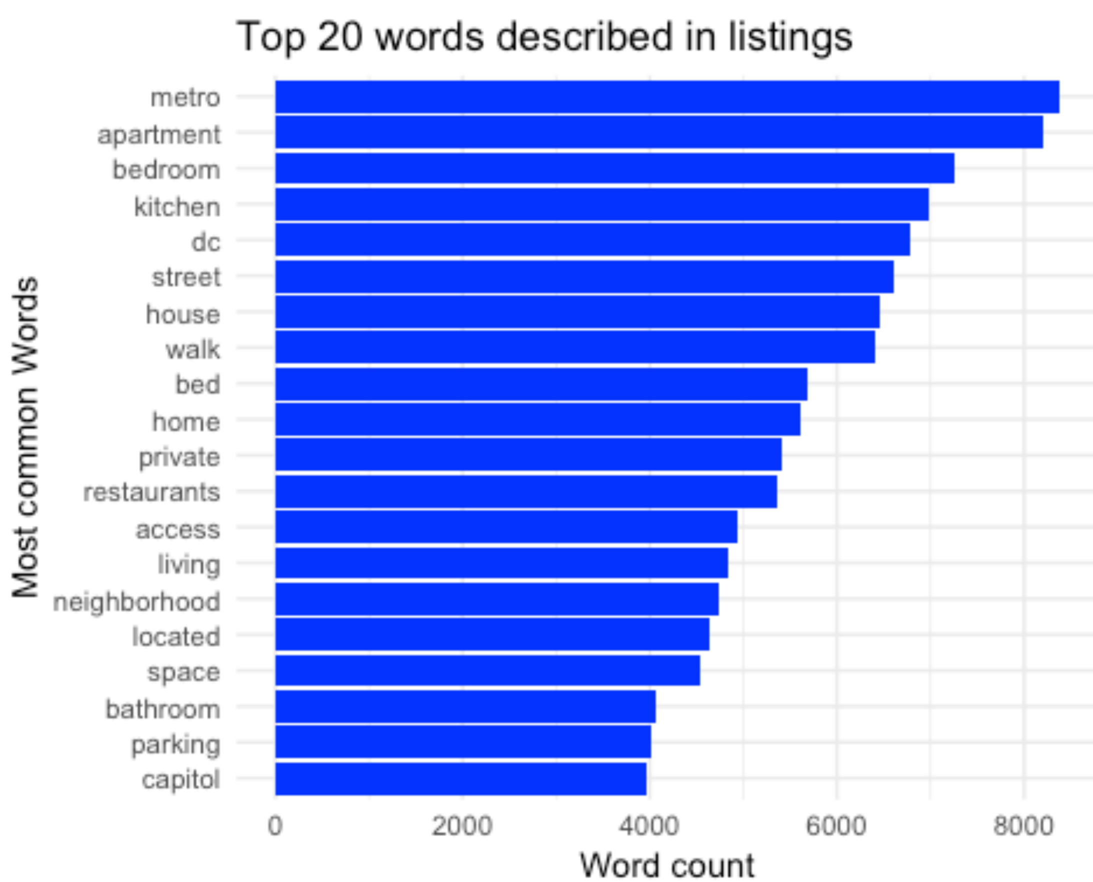
Most Frequent words used by hosts in descriptions

```
# unnest_tokens function to tokenise
listings_words <- listings %>%
  select(id, description, price, review_scores_accuracy,
review_scores_rating) %>%
  unnest_tokens(word, description) %>%
  filter(!word %in% stop_words$word,
```

```
        str_detect(word, "^[a-z']+$"))

#plot the graph
common_listings <- listings_words %>%
  group_by(word) %>%
  summarise(count = n()) %>%
  top_n(n = 20, wt = count) %>%
  ggplot() +
  geom_bar(mapping = aes(x=reorder(word, count),
                         y=count),
           stat="identity", fill = "blue") +
  labs(title="Top 20 words described in listings",
       y="Word count", x="Most common Words") +
  coord_flip() +
  theme_minimal()

common_listings
```

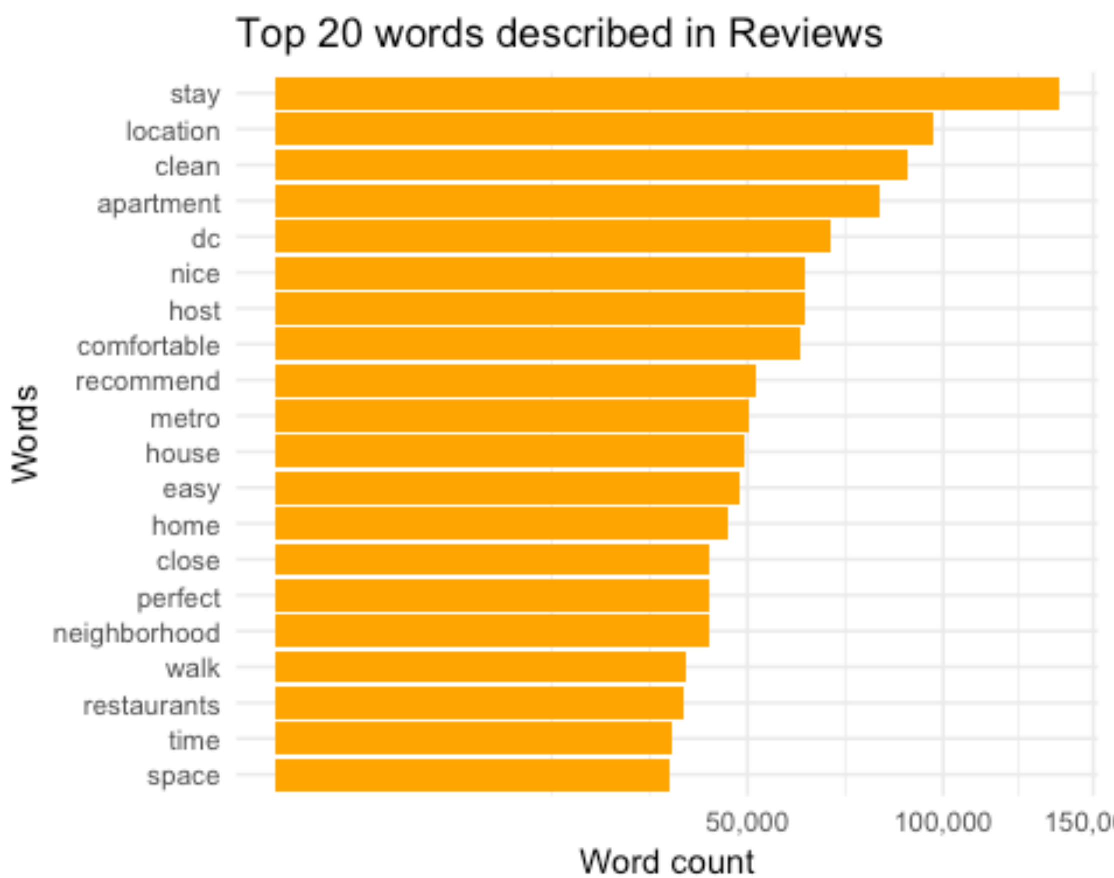


From the plot above of top 20 words in listing descriptions, it seems like most of the hosts mention about the proximity to metro in the description.

Most Frequent words used by guests in reviews

```
# Using unnest_tokens function to takenise
review_words <- reviews %>%
  unnest_tokens(word, comments) %>%
  filter(!word %in% stop_words$word,
         str_detect(word, "^[a-z']+$"))

op <- par(mar = c(9,4,4,2) + 0.1)
#plot the graph
common_reviews <- review_words %>%
  group_by(word) %>%
  summarise(count = n()) %>%
  top_n(n = 20, wt = count) %>%
  ggplot() +
  geom_bar(mapping = aes(x=reorder(word, count), y=count),
           stat="identity", fill = "orange") +
  coord_flip() +
  labs(title="Top 20 words described in Reviews",
       y="Word count", x="Words") +
  theme_minimal()+ scale_y_sqrt(labels = scales::comma)
common_reviews
```




```
par(op)
```

From the above plot of top 20 keywords in reviews, it seems like stay, location, clean, host, comfort are some of the important factors that matter to the guests.

Neighbourhood

```
factpal <- colorFactor(topo.colors(3), listings$neighbourhood_cleansed)

popup <- paste0("<strong>'hood: </strong>", listings$neighbourhood_cleansed)

leaflet(listings) %>% addProviderTiles("CartoDB.DarkMatter") %>%
  addCircleMarkers(
    color = ~factpal(neighbourhood_cleansed),
    stroke = FALSE, fillOpacity = 0.5, radius = 1.2,
    popup = ~popup
  )
```

```
Assuming "longitude" and "latitude" are longitude and latitude, respectively
```

```
#Finding the count by group_by and sort
listing_groupby<- listings %>%
  group_by(neighbourhood_cleansed)
count_list <- count(listing_groupby, sort = TRUE)
kable(count_list)
```

neighbourhood_cleansed	n
Columbia Heights, Mt. Pleasant, Pleasant Plains, Park View	910
Union Station, Stanton Park, Kingman Park	906
Capitol Hill, Lincoln Park	858
Edgewood, Bloomingdale, Truxton Circle, Eckington	713
Dupont Circle, Connecticut Avenue/K Street	685
Shaw, Logan Circle	623
Downtown, Chinatown, Penn Quarters, Mount Vernon Square, North Capitol Street	499
Brightwood Park, Crestwood, Petworth	477
Kalorama Heights, Adams Morgan, Lanier Heights	423
Howard University, Le Droit Park, Cardozo/Shaw	362
West End, Foggy Bottom, GWU	350
Georgetown, Burleith/Hillandale	284
Ivy City, Arboretum, Trinidad, Carver Langston	245
Takoma, Brightwood, Manor Park	165
Brookland, Brentwood, Langdon	159
Southwest Employment Area, Southwest/Waterfront, Fort McNair, Buzzard Point	150
Cathedral Heights, McLean Gardens, Glover Park	140
Cleveland Park, Woodley Park, Massachusetts Avenue Heights, Woodland-Normanstone Terrace	129

Lamont Riggs, Queens Chapel, Fort Totten, Pleasant Hill	109
Twining, Fairlawn, Randle Highlands, Penn Branch, Fort Davis Park, Fort Dupont	109
Spring Valley, Palisades, Wesley Heights, Foxhall Crescent, Foxhall Village, Georgetown Reservoir	98
Friendship Heights, American University Park, Tenleytown	96
Congress Heights, Bellevue, Washington Highlands	86
North Michigan Park, Michigan Park, University Heights	84
North Cleveland Park, Forest Hills, Van Ness	83
Capitol View, Marshall Heights, Benning Heights	79
Near Southeast, Navy Yard	74
Woodridge, Fort Lincoln, Gateway	68
Hawthorne, Barnaby Woods, Chevy Chase	56
Mayfair, Hillbrook, Mahaning Heights	53
Historic Anacostia	50
Colonial Village, Shepherd Park, North Portal Estates	47
Sheridan, Barry Farm, Buena Vista	46
Deanwood, Burrville, Grant Park, Lincoln Heights, Fairmont Heights	42
River Terrace, Benning, Greenway, Dupont Park	40
Douglas, Shipley Terrace	28
Fairfax Village, Naylor Gardens, Hillcrest, Summit Park	21
Woodland/Fort Stanton, Garfield Heights, Knox Hill	13
Eastland Gardens, Kenilworth	9

From the plot and table above, we see that most number of listings are close to the neighbourhoods Columbia Heights, Union stations, Capitol Hill etc.

Review score rating

```r
review_desc <- listings$review_scores_rating
summary(review_desc )
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
  20.00   93.00   97.00   94.96  100.00  100.00    2214 
```

As seen above, most of the guest who review give high scores.

Different listings based on Room type

```r
room_groupby<- listings %>%
  group_by(room_type)
count_room <- count(room_groupby, sort = TRUE)
kable(count_room)
```

room_type	n

Entire home/apt	6614
Private room	2525
Shared room	230

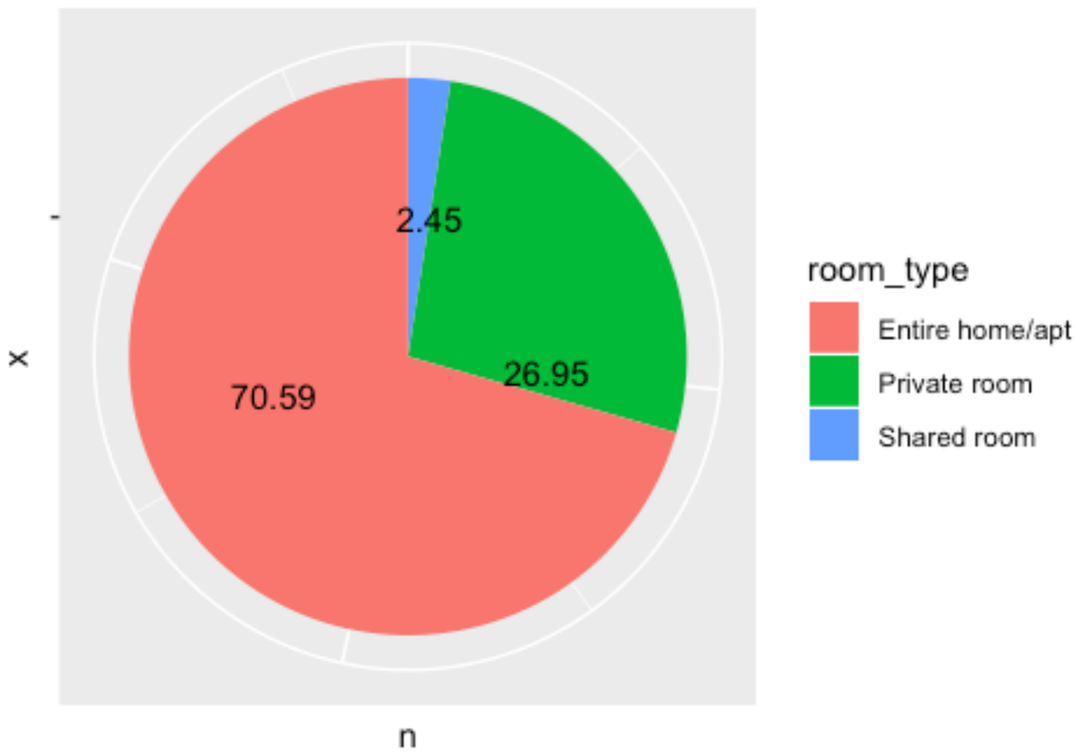
Property Type of listings

```
listings$property_type = ifelse(listings$property_type == "Apartment",
"Apartment",
                           ifelse(listings$property_type == "Bed &
Breakfast","B&B",
                           ifelse(listings$property_type ==
"Condominium","Condominium",
                           ifelse(listings$property_type == "House","House",
                           ifelse(listings$property_type == "Loft","Loft",
                           ifelse(listings$property_type ==
"Townhouse","Townhouse",
                           ifelse(listings$property_type == "Dorm","Dorm",
"Other")))))))
listings$property_type = as.factor(listings$property_type)
```

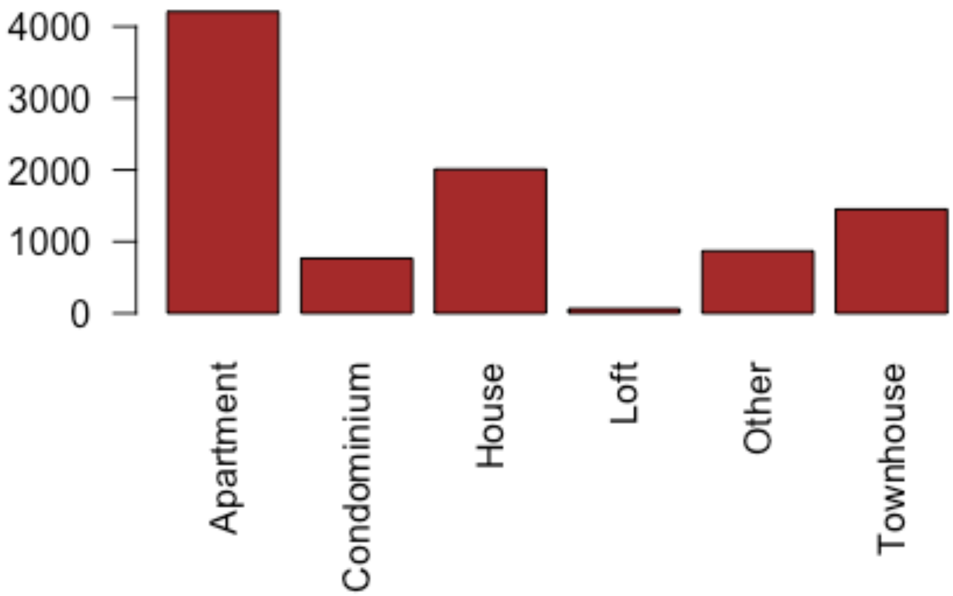
We have only kept Apartment, b&b, Condominium, House, Loft, Townhouse, and Dorm in Property.Type and the rest would be categorised to Others.

```
s <- unique(listings$property_type)

cr <- data.frame(count_room)
piepercent<- round(100*(cr$n/sum(cr$n)),2)
bp<- ggplot(cr, aes(x="", y=n, fill=room_type))+
geom_bar(width = 1, stat = "identity")
pie <- bp +
coord_polar("y")+geom_text(label=piepercent)+theme(axis.text.x=element_blank(
))
pie
```




```
op <- par(mar = c(9,4,4,2) + 0.1)
barplot(table(listings$property_type),col = "brown",las=2)
```




```r
par(op)
```

The above pie chart shows people prefer entire property than private rooms or shared rooms. On exploring further about property type, it is seen that majority of the listings are Apartment, House, townhouse.