Project 1 Writeup

Introduction

Figuring out where you will be staying is an important part of traveling and is often the largest cost in your travel budget. In this analysis, I break down a dataset containing listings of Airbnb properties in Asheville, NC to determine what characteristics predict the price of a listing. My goal is to create a predictive model to determine if I'm being overcharged for the listing I chose to use for my honeymoon.

Dataset Description

This dataset was sourced from InsideAirbnb.com in August of 2020. The last time this dataset had been updated was on June 25, 2020. It contains 2,407 different listings with 106 points of data on each property. Examples of included properties are superhost status, price, bedrooms, zipcode, and rating.

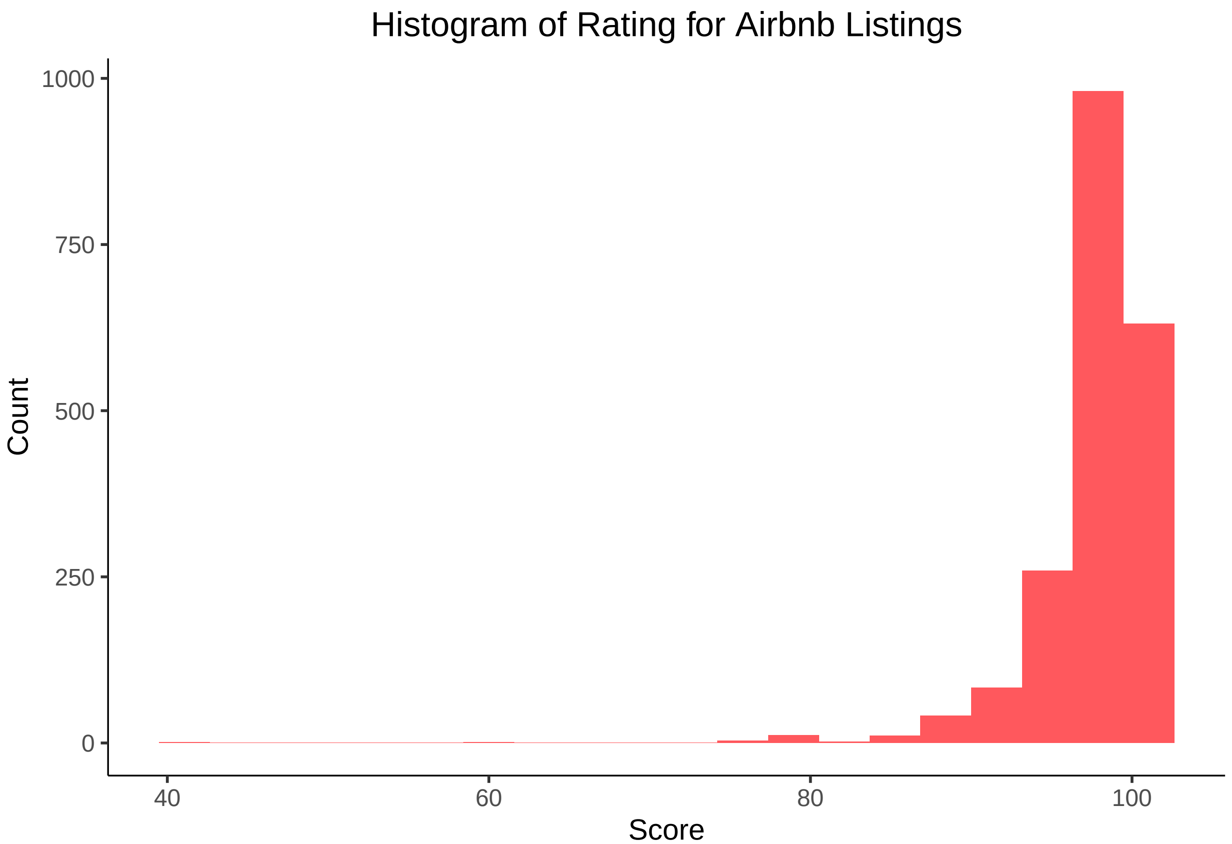
Analysis

I first explore my outcome variable, price. Price is represented in multiple ways in the dataset including ???, but I determined to look at the price for a one-night stay, including the cleaning fee. Thus, the usage of "price" in this analysis represents the sum of rent for one night and the cleaning fee.

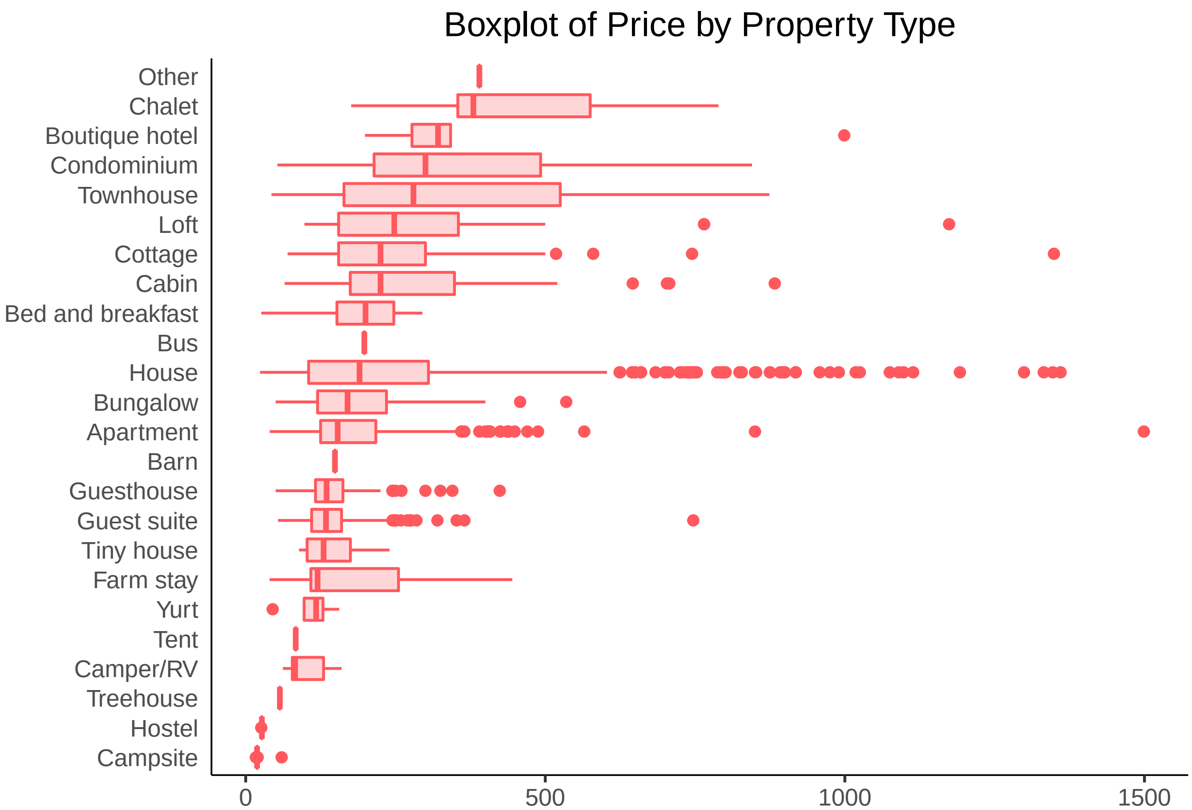
Since the goal of my analysis is to create a model that can be used to predict prices as I browse Airbnb, I am limited to predictors that can be determined based on the listing page alone. I determined that the easiest predictors that are both available in the dataset and available from the listing page is the property type, the room type, the number of bathrooms, the number of bedrooms, the number of beds, the average score of reviews, and if the host has superhost status.

Exploration

```
## Warning: Removed 177 rows containing non-finite values (stat_bin).
```



As we can see in the histogram above, the scores are skewed to the left and have a tight distribution, making it difficult to distinguish listings based on rating alone. Thus, I decided to eliminate rating as a potential predictor.



Due to the great variation in price by property type, I decided to narrow my analysis to a smaller scope and only examined types of properties that we would potentially book: condominiums, townhomes, houses, and apartments.

Analysis Dataset

After limiting the dataset to relevant property types and to the predictors we are interested in, the data now has 1,419 complete cases for us to predict and test on.

We split the data about 50/50 to create testing and training datasets using a seed of 1107.

Best subset selection

```
regfit.best=regsubsets(total_price~.,train,nvmax=9)
test.mat=model.matrix(total_price~.,test)
val.errors=rep(NA,5)
for(i in 1:5) {
  coefi=coef(regfit.best,id=i)
  pred=test.mat[,names(coefi)]%*%coefi
  val.errors[i]=mean((test$total_price-pred)^2)
}
num=which.min(val.errors)
coef(regfit.best,num)
```

```
##             (Intercept) property_typeCondominium    room_typePrivate room
##                33.34688                149.91815                -79.67349
##               bathrooms                     beds         host_is_superhostt
##               102.81855                 34.71874                -27.21339
```

```r
val.errors[num]
```

```
## [1] 14052.95
```

Forward selection

```r
regfit.best=regsubsets(total_price~.,analysis,nvmax=9,method="forward")
test.mat=model.matrix(total_price~.,test)
val.errors=rep(NA,7)
for(i in 1:7) {
  coefi=coef(regfit.best,id=i)
  pred=test.mat[,names(coefi)]%*%coefi
  val.errors[i]=mean((test$total_price-pred)^2)
}
num=which.min(val.errors)
coef(regfit.best,num)
```

```
##             (Intercept) property_typeCondominium      property_typeHouse
##                32.47772                126.58808               -19.08408
##      room_typePrivate room                bathrooms                bedrooms
##               -69.27565                107.42136                20.15791
##                    beds        host_is_superhostt
##                19.22847                -24.70380
```

```r
val.errors[num]
```

```
## [1] 13393.39
```

Backward selection

```r
regfit.best=regsubsets(total_price~.,analysis,nvmax=9,method="backward")
test.mat=model.matrix(total_price~.,test)
val.errors=rep(NA,7)
for(i in 1:7) {
  coefi=coef(regfit.best,id=i)
  pred=test.mat[,names(coefi)]%*%coefi
  val.errors[i]=mean((test$total_price-pred)^2)
}
num=which.min(val.errors)
coef(regfit.best,num)
```

```
##             (Intercept) property_typeCondominium      property_typeHouse
##                32.47772                126.58808               -19.08408
##      room_typePrivate room                bathrooms                bedrooms
##               -69.27565                107.42136                20.15791
##                    beds        host_is_superhostt
##                19.22847                -24.70380
```

```r
val.errors[num]
```

```
## [1] 13393.39
```

It appears the model selected by forward and backward selection are the exact same. Since the error rate is lower for this model compared to the model selected by best subset selection, we shall use this as our predictive model.

Prediction

```
coefi=coef(regfit.best,id=num)
sample=c(1,0,1,1,1,1,1,1)
coefi%*%sample
```

```
##          [,1]
## [1,] 66.22193
```

The predicted price of the linked listing is $66.22 which is much lower than the price which is $175.

Listing used for prediction: https://www.airbnb.com/rooms/34151081