Predicting Airbnb Listing Price and Satisfaction

DSBA 6211 Advanced Analytics

May 11, 2021

Elijah Ellison | Ivan Flores Martinez | Bailey Brown | Ryon Ripper | Tristan Richard

**1 Introduction and Goal**

Airbnb is a unique platform that benefits both travelers and the hosts. In this project, our goal was to analyze the listing price of homes in the Asheville area as well as predict if a listing will have a high satisfaction rating. The prices are typically determined by the hosts, which is different from hotels because you can rent out an entire house instead of just one room. It also allows larger group trips to be made more affordable in an Airbnb stay because there is generally no per-person fee that must be paid.

Creating a fair price that is a win-win scenario for both the travelers and hosts is a challenge. However, what we were able to do is predict prices based on the distance from the center of the city, using various machine learning approaches. Another variable of interest was overall satisfaction, which is a rating given to a listing by travelers staying at the property. Using the overall rating of other listings in the Asheville area, we predicted whether or not a property will get a high satisfaction rating based on various predictor variables.

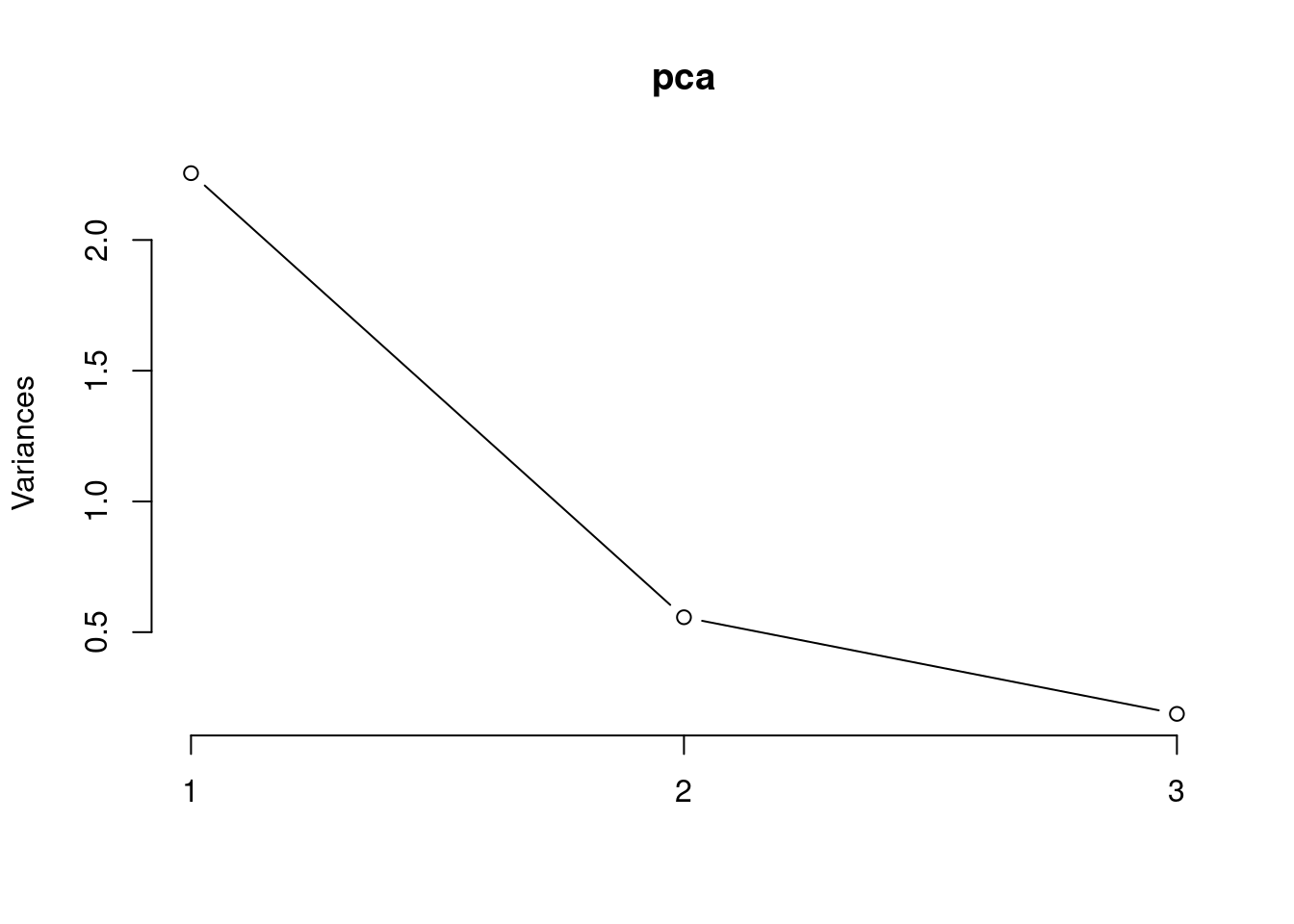
This would be useful for anyone who is looking to invest in a rental property in the Asheville area because using the models, you could see if a potential listing you are interested in purchasing would be received well by the users of Airbnb. Using our model to determine if a property is worth buying could help investors tremendously in their decision making process.

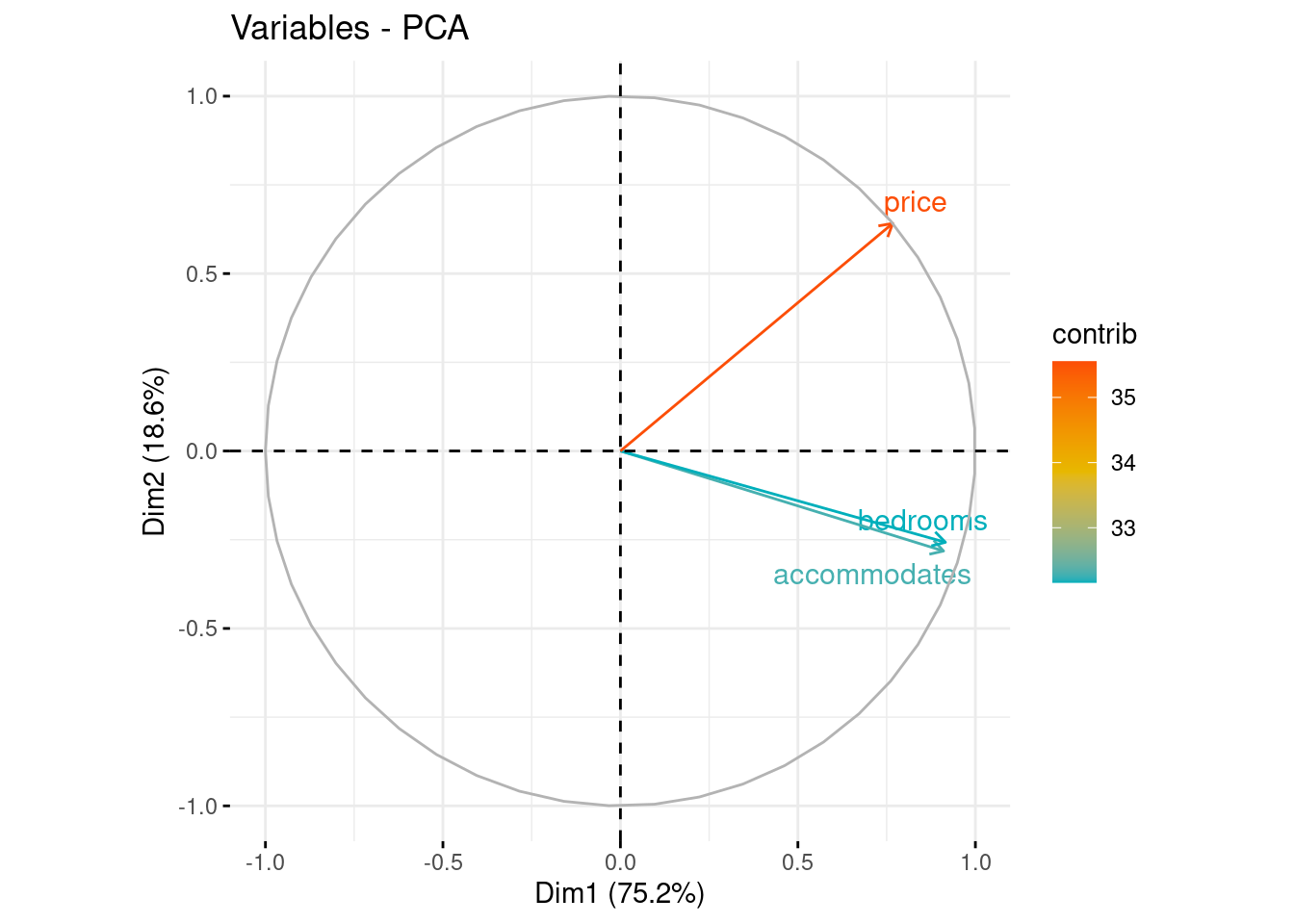
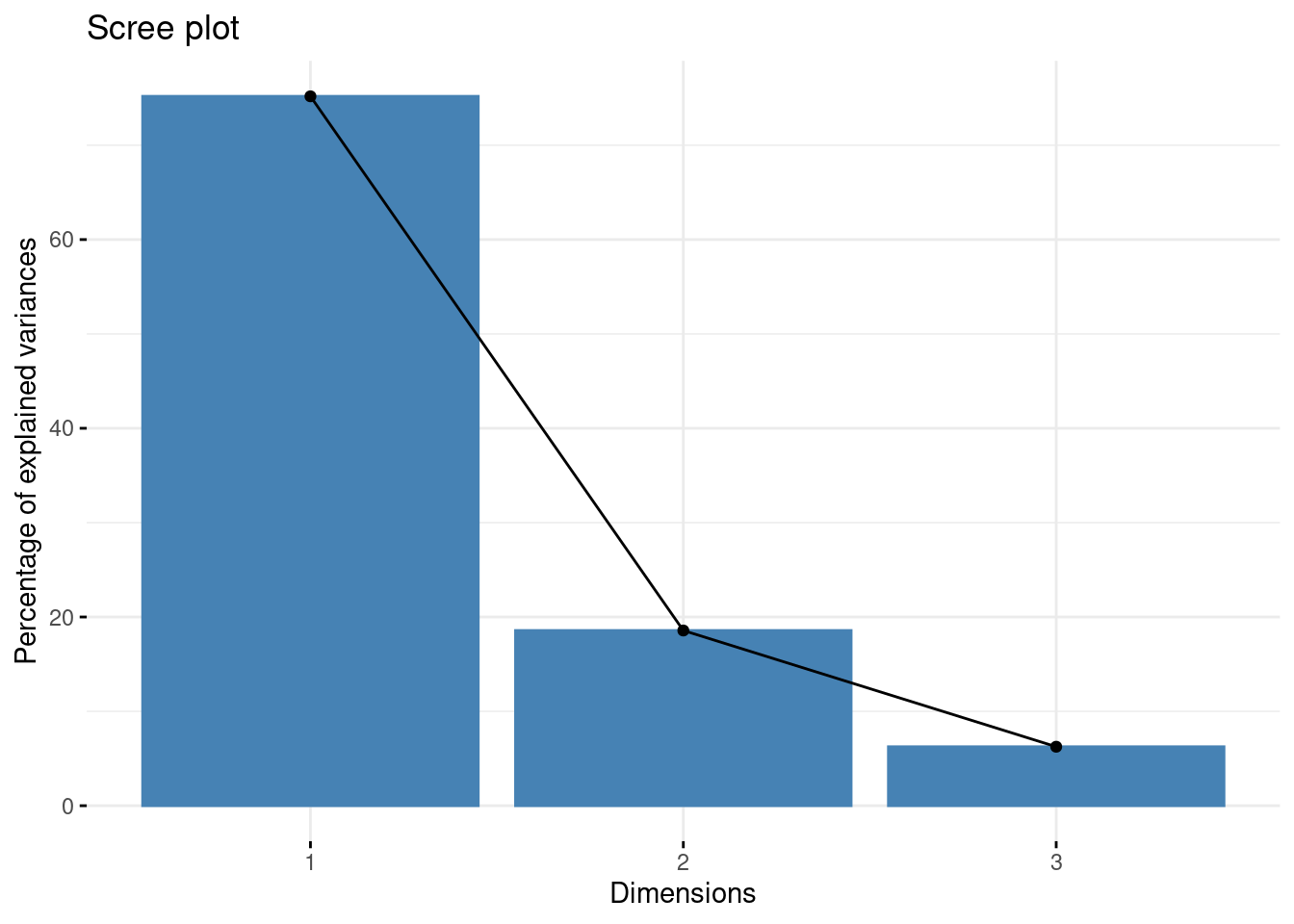
Another useful aspect of our project would be to help boost the satisfaction rating of existing Airbnb listings by using the model and adjusting the price and other attributes in order to get the model to predict the listing as having a high rating. Thus, hopefully improving your existing Airbnb property’s satisfaction rating. Our goal for this project was to make effective models that predict prices and travellers satisfaction of Asheville Airbnb listings so that the models could be used to help investors decide if purchasing a particular property in Asheville would be a worthwhile investment, as well as help travellers determine if a listing's price is fair.

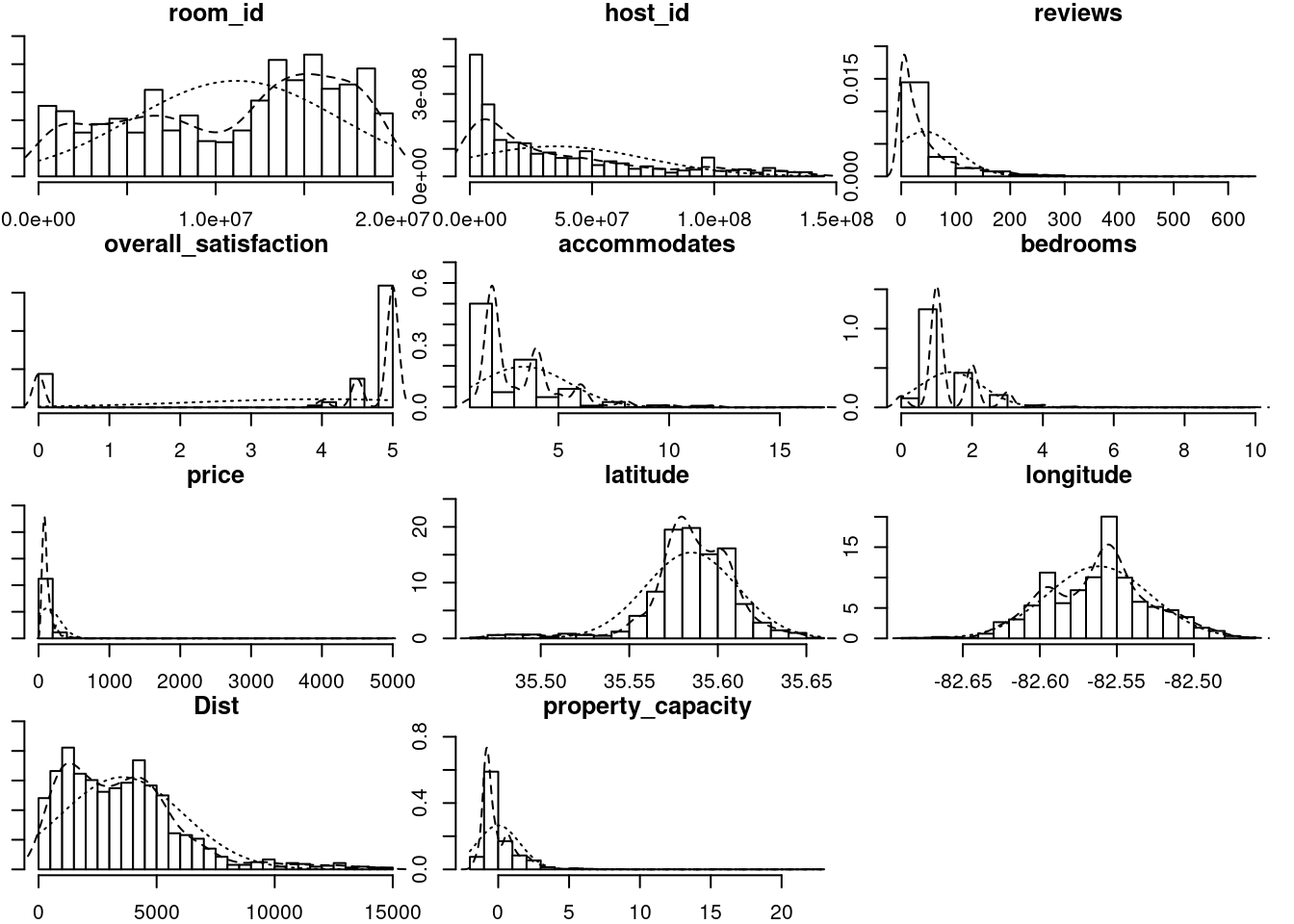
**2 Data Cleaning and Processing**

The Airbnb data that we used were in multiple excel files that were all surveys previously done by Airbnb. First, we downloaded all of the data sets and combined them into a merged data set with the use of R. Then, we went through all of the attributes of the merged data set and removed variables that had a lot of null values, variables that had the same value for all the listings, and variables that didn’t hold much information for our analysis on the data. These variables were borough, mainstay, country, bathrooms, survey\_id, location, and city. We also removed any listings that were potentially duplicated when the data sets were merged so that each listing was unique. After this step, we proceeded by checking the variables to see which ones had missing values that needed to be imputed, and we found that the overall\_satisfaction variable had a couple of missing values, so we used mean imputation to impute these values.

After the initial preprocessing of our data, we created a reference point for the center of the downtown Asheville area using the longitude and latitude, which we then used as the end point for our distance calculation. Using functions in R, we calculated the distance between the longitude and latitude of each of the listings to the longitude and latitude of the downtown reference point, and created a new distance to downtown variable to use for analysis. Next, the overall satisfaction variable only had values for satisfaction that were either really high or really low, so we decided to create a new variable from the overall satisfaction variable called high satisfaction. This binary variable indicates whether the listing has a satisfaction at or above a 4.0 star rating, which we will be using for our analysis.

Once these new variables were created, we needed to make sure any variables that were highly correlated were not going to affect our models, so we used PCA to check and create PC variables. We found that the variables “bedrooms” and “accommodates” were highly correlated (0.81). Consequently, we ran PCA to reduce the multicollinearity. Below are some of the charts that were generated when we performed the PCA. 



We only used the first PCA in our model because it captured more than 75% of the variance, and named it “property capacity”. Following the PCA, we evaluated the distribution of the numeric variables in our dataset to make sure there were no variables that were highly skewed or highly skilled. We determined from the graphs that price, reviews, and distance were all fairly skewed, so we decided to scale the variables to fix this issue. Here is the plot with the original distribution of the variables:We then created dummy variables for categorical variables with more than 2 categories, and we standardized the remaining numerical variables. Finally, we removed variables that only added noise, and also removed any outliers that would negatively affect our models (rooms with prices over $600 USD).

**3 Model Analysis and Performance**

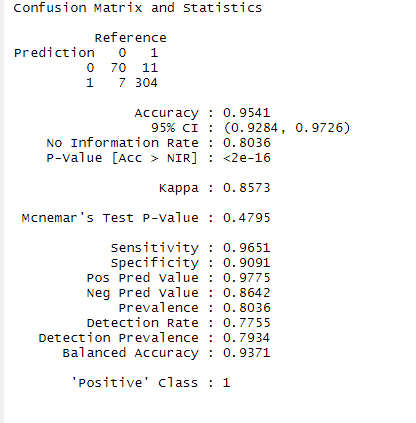
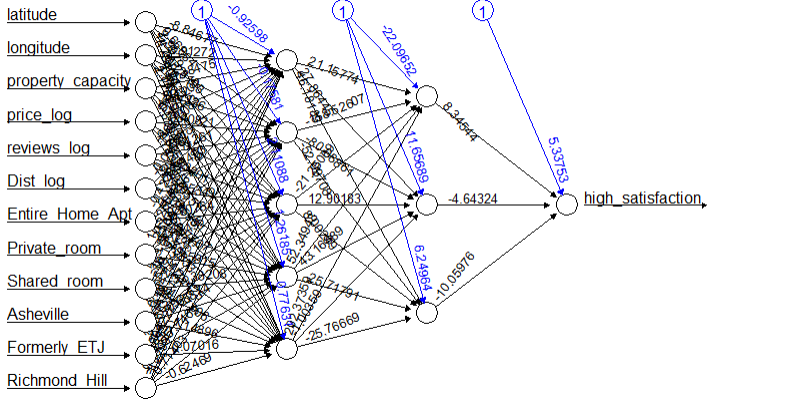
For the model analysis and performance portion of our assignment, we created multiple models to predict one of two things: either the price of a listing, or if a listing would receive a high satisfaction rating. We mainly focused our energy on predicting the price, and we created 6 different models to try to predict the price of listings. For predicting the high satisfaction variable, we only created 3 models to try to predict the outcome.

The models used in order to predict if a listing would receive a high satisfaction from a traveler are shown below:

**Neural Network Model:**

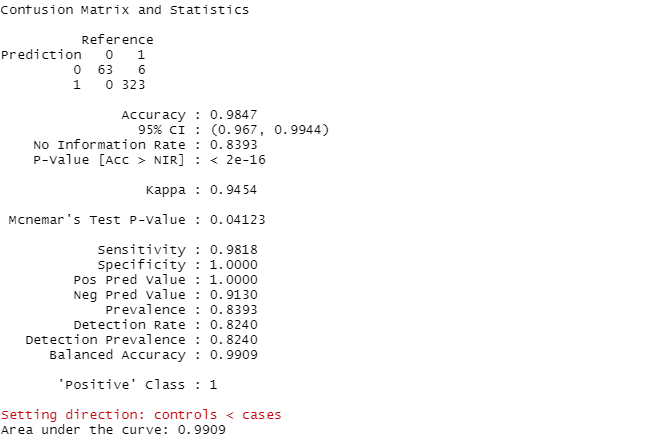
To use this model, we first removed the last modified variable from our dataset because we felt that it did not have much relevance in this prediction. Then, we factored the high satisfaction variable so that the model could properly read and predict the variable in R. Next, we used the neuralnet() function to train the model to predict the high satisfaction rating on the training dataset, and after that we used the predict() function to predict the high satisfaction on the testing dataset.

The results for the model are as follows:



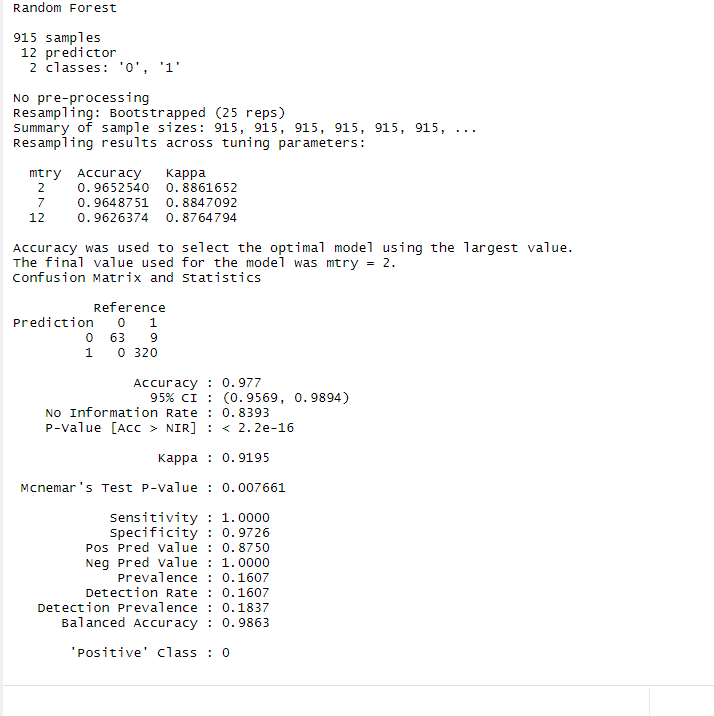
**Decision Tree Model:**

For the decision tree model, all we needed to do to the variables was factoring all the dummy variables in order to make a decision tree. After this, we created a training and testing data set and built the tree model. Here are the results of the decision tree:



**Random Forest Model:**

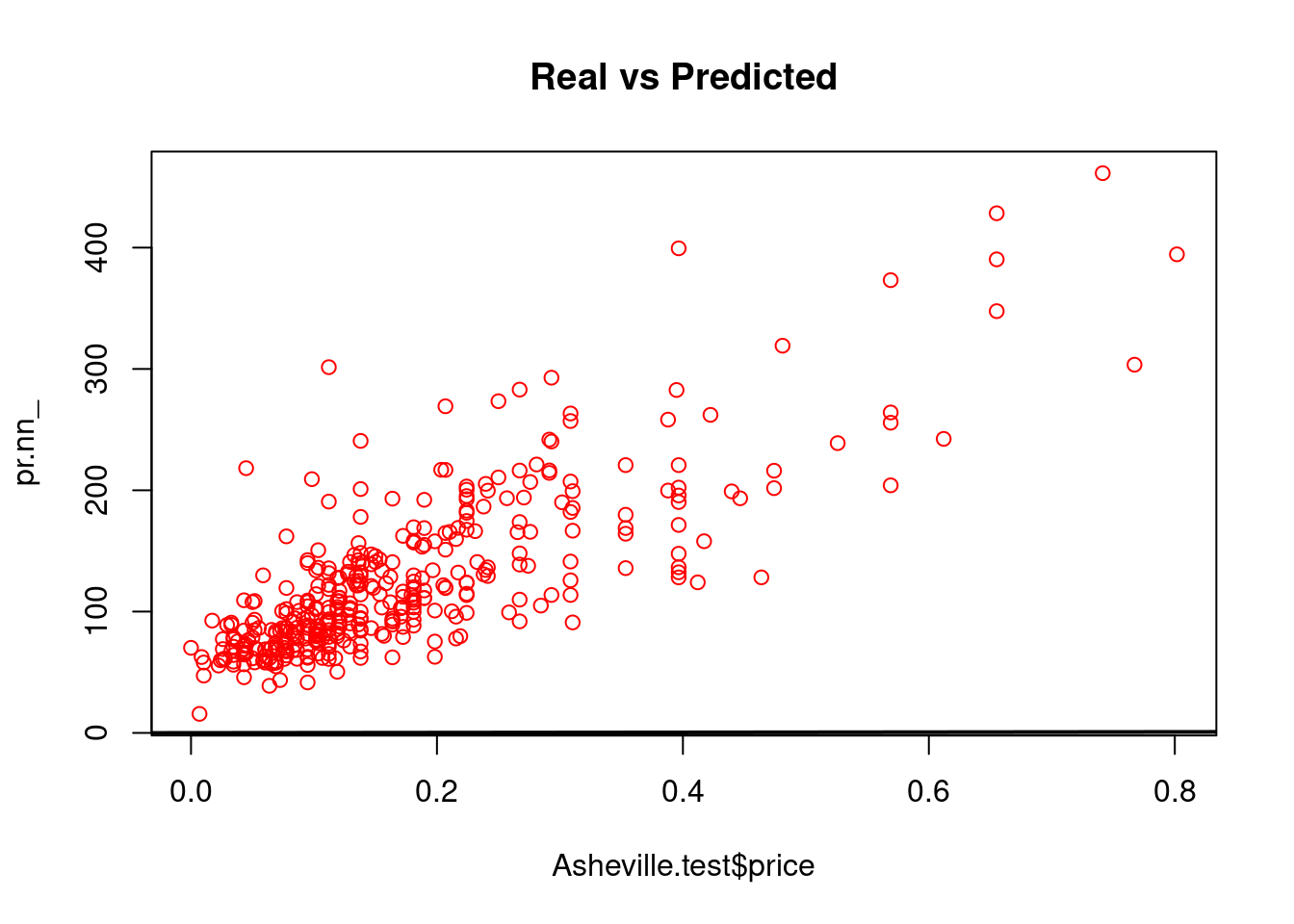
Since all of the data preprocessing was done by the last two models, all we needed to do was use the train() function in R to predict high satisfaction of a listing from the training data set based on accuracy for the random forest model. Then, we used the trained model to predict if a listing had a high satisfaction on the testing data set. Here are the results from the model:



Models used in order to predict the price of a listing are show below:

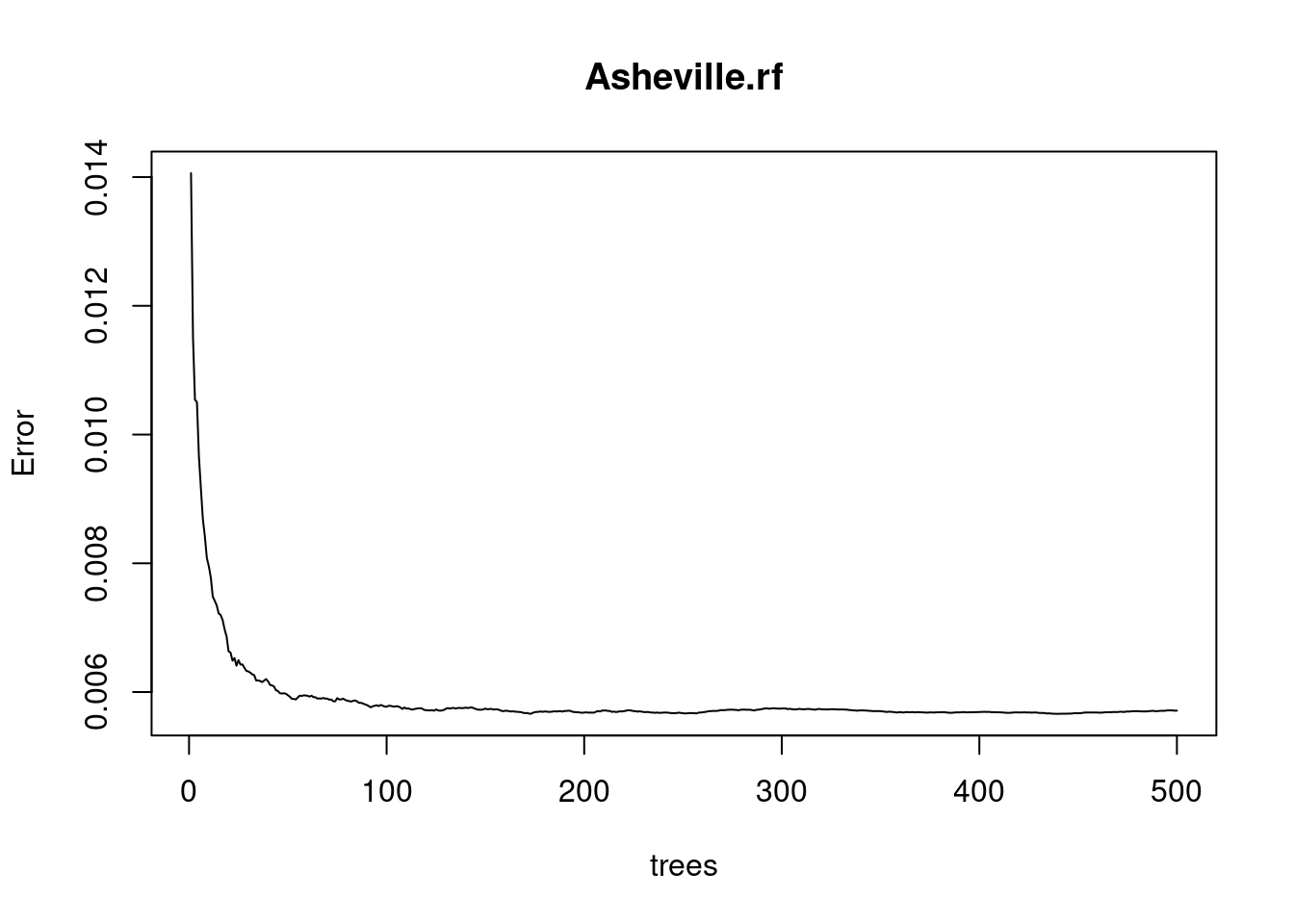
**Neural Network:**

We trained the Neural Network Model using the neuralnetwork() function with price as the outcome variable and reviews, high\_satisfaction, latitude, longitude, property\_capacity, Dist, Richmond\_Hill, Formerly\_ETJ, Private Room, Entire\_Home\_Apt as the predictors. The neural network that performed best had one hidden layer with 2 units. When we tried models with a higher number of hidden layers/units the model overfitted to the training dataset. Running the model with the optimal neural architecture resulted in .08015 for the RMSE, and .621 for the R Squared Value.

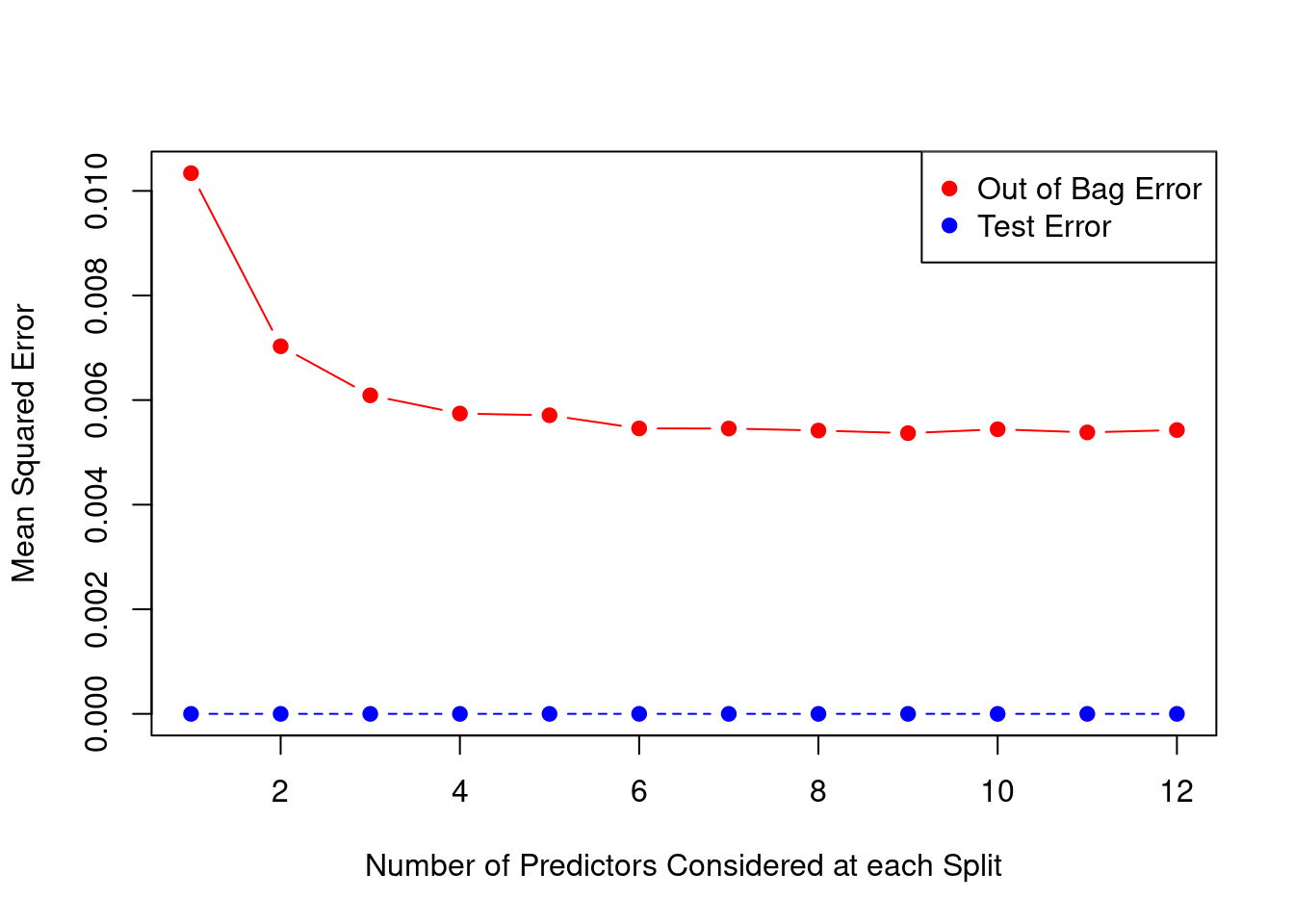
****

**Random Forest:**

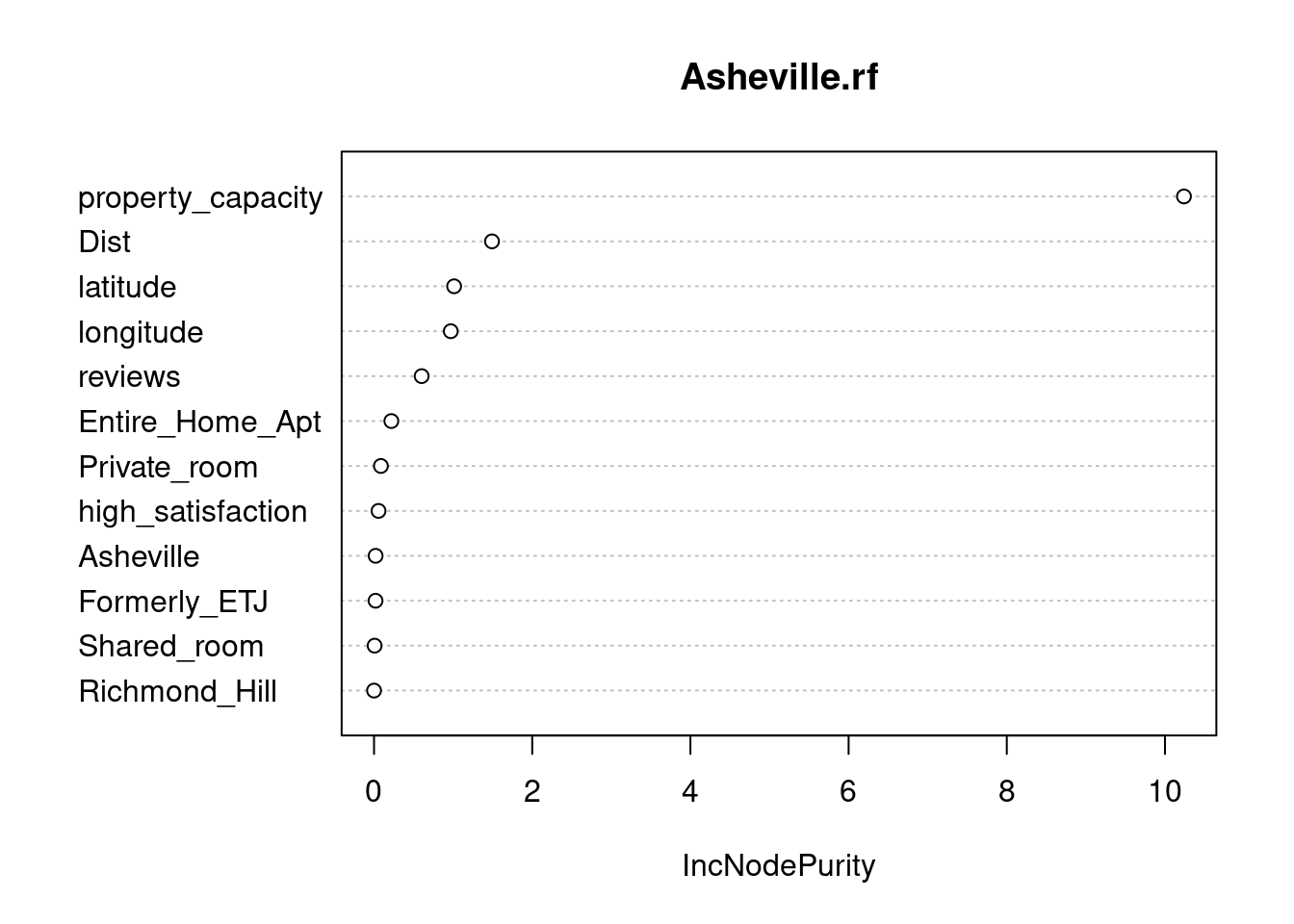
Since the data was already preprocessed, we simply used the randomforest() function in R and came up with a RMSE of .07355 and a R Squared Value of .7079. Below are the graphs used in our random forest analysis.

Here is the error per number of trees:

MSE for each number of predictors at each split:



Variable Importance in our Random Forest Model:

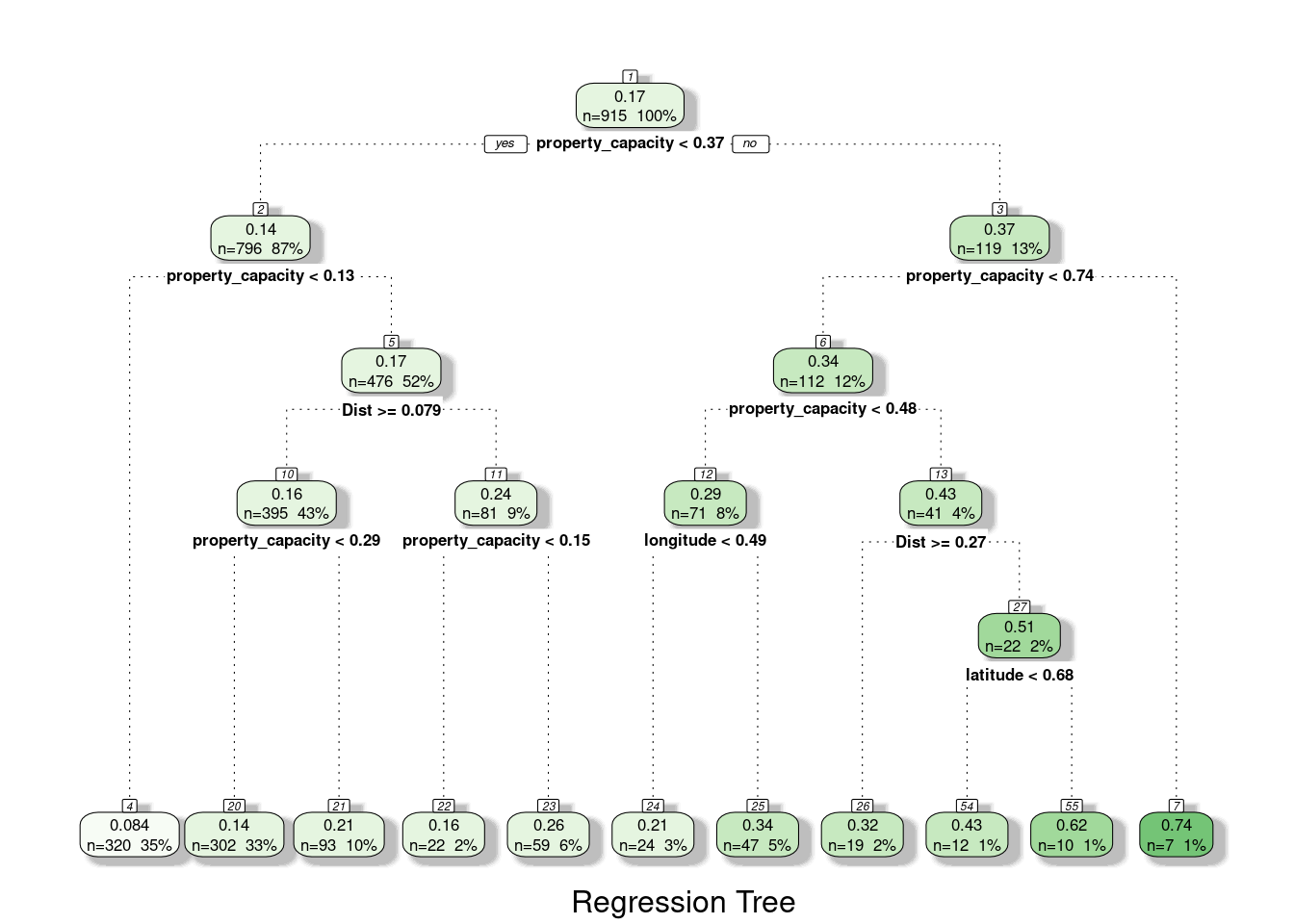


**KNN:**

For this model, we first needed to choose a value for k in knn. For the K value, we decided to minimize the MSE for all k values between 1 and 30. We found that the value of k that minimized the MSE was 12. Once we figured out what K to use we ran the KNN model in R and found that we got a RMSE of 0.08451375 and a R Squared value of 0.5785441

**Decision Tree:**

For the decision tree, we used the rpart() function in R with the method set to anova. From there, we calculated the regression tree and got an RMSE of .09036 and a R Squared Value of .5182. Here is the tree we got from the model:



**Boosted Decision Tree:**

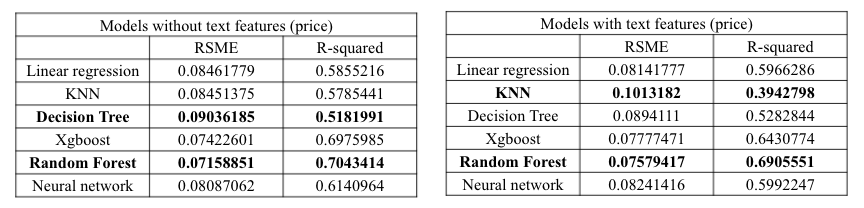
For this model, we trained the xgbtree using a tune grid with nrounds=50, max depth=5, eta=0.1, gamma=0, colsample by tree=0.9, min child weight=1, subsample=1. After training and running the model, we found the RMSE to be .07159 and a R Squared value of .6976.

**Linear Regression:**

For the linear regression model, we used the glm() function with price as the outcome variable and reviews, high satisfaction, latitude, longitude, property capacity, distance, Richmond Hill, Formerly ETJ, Private Room, and Entire Home Apt as the predictor variables. For this model we found the RMSE to be .08462 and the R Squared to be .5855

**Text Mining:**

Text mining was briefly implemented in order to analyze the “name” variable in our dataset, which contains the title for each listing. Text features from a unigram were included in each of our models, and the results of each model with and without the text features are shown below:



We can see that the random forest model was the most accurate at predicting the listing price of an Airbnb. The decision tree was the model that performed the worst without the text features, and the KNN was the model that performed the worst with the text features. Simpler models improved with text features (Linear Regression and Decision Tree), while the more complex models performed worse with text features because they overfitted to the training dataset (NN, Random Forest, XGboost).

**4 Additional Suggestions and Recommendations**

After completing the project and reflecting on our processes and results, there are a couple of things that we would’ve liked to add in if we were given more time. Our Airbnb data didn’t really have a variable for determining how profitable a listing is in the year overall. This is a big piece of information that would have changed a lot for our project. Having a variable for either average annual income or average nights occupied by day or month or year would have been some great information to add to our dataset. We could have predicted the average annual income for potential new listings, or determined how busy a listing would be given certain parameters, which would be very informative for investors. We could’ve used those parameters to better our own models performance as well.

We also would have liked to add in a variable to show some sort of seasonality for the listings, like how many nights the listing is occupied in the spring, summer, fall, or winter, for example. This could have helped with model accuracy, as well as good variables to look at for potential investors for certain listings, because it shows how busy a listing is for each season. Another variable that we would have liked to add in that would be similar to a seasonality variable would be a variable showing how many consecutive nights or weekends any listing would be booked for. This is also another very informative variable we would have liked to add in to better our models accuracy.

Finally we would have liked to further utilize text mining techniques in our analysis. We were able to implement a unigram text analysis in our models, however, there are ways we could have improved. We began creating a bigram for better analysis but were unable to add the text features into our models due to time constraints. In order to provide insight to the Airbnb hosts about specific words or phrases that could boost their listings, further text analysis such as bigrams, unigrams + bigrams, etc. should be implemented in the future.

**5 Summary and Conclusion**

Our goal for this project was to make effective models that predict prices and traveler satisfaction of Asheville Airbnb listings, which could be used to help investors decide if purchasing a particular property in Asheville would be a worthwhile investment, or help travellers look for Airbnbs that are well priced. We spent a lot of time cleaning and preprocessing all of the raw airbnb data down to a data set that is serviceable for our models to run and perform with high accuracy.

We then partitioned the data set, trained and tested our models, and produced models that had the best performance and accuracy. We found when predicting if a listing would have high satisfaction, that the random forest model was very high in accuracy, at about 97%, and we thought it was more robust at predicting high satisfaction than the decision tree model. For the models used to predict price, we found most of them to be okay at prediction, but some of them performed better than others. Mainly, the boosted decision tree and the random forest model performing the best out of all of them. The boosted decision tree had an RMSE of .07159 and an R-squared value of .6976 while the random forest model had an RMSE of .07355 and an R-squared Value of .7079. As you can see, both are very similar when it comes to the parameters they are assessed by, and both of them perform really well to help potential users of the models properly predict price.

In conclusion, we can use the models created to predict if a listing has a high satisfaction to see if potential listings in the Asheville area would be worth the investment. The models could also be used to determine how to get an existing low-rating Asheville listing to a higher satisfaction by adjusting things like price on your listing to try to get the model to predict a high satisfaction. You can also use the models we have created to find out the price of any listing in the Asheville area to make sure that your Airbnb is priced correctly if you’re a landlord or to see if you are getting a good deal if you are the traveller planning to stay at an Airbnb. There are lots of applications for the models we have created and we hope that they could be of some use to those who want to use them.

**Data Sources:**

<http://insideairbnb.com/behind.html>

Zip file for Asheville data downloaded here:

<http://tomslee.net/category/airbnb-data>