Airbnb in Asheville, NC

By: Tristan Richard, Bailey Brown, Ivan Martinez, Ryon Ripper and Elijah Ellison

May 11th, 2021

DSBA 6211 Advanced Business Analytics

Project Overview

01 Introduction

02 Data cleaning and preprocessing

03 Analysis Methods & Objectives

04 Model Performance

05 Future Work

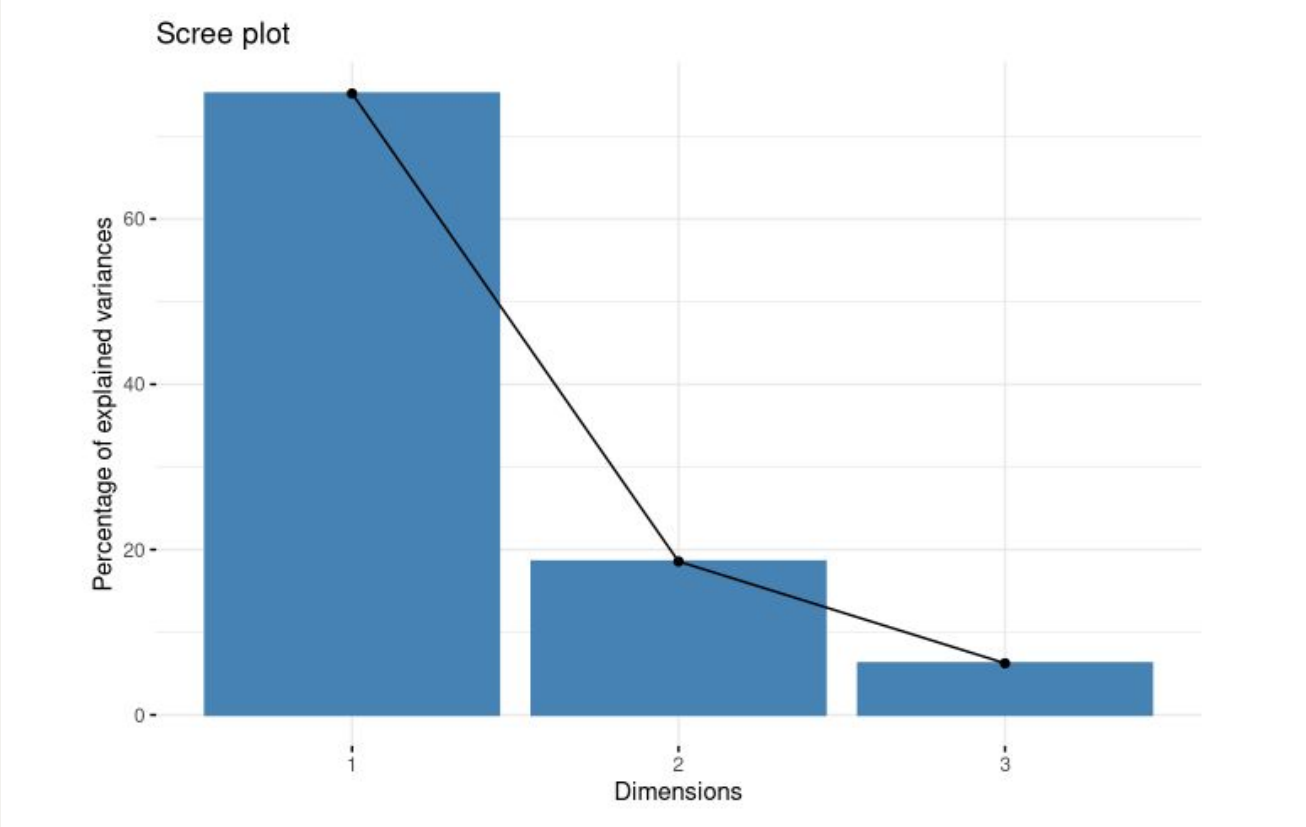
01 Introduction

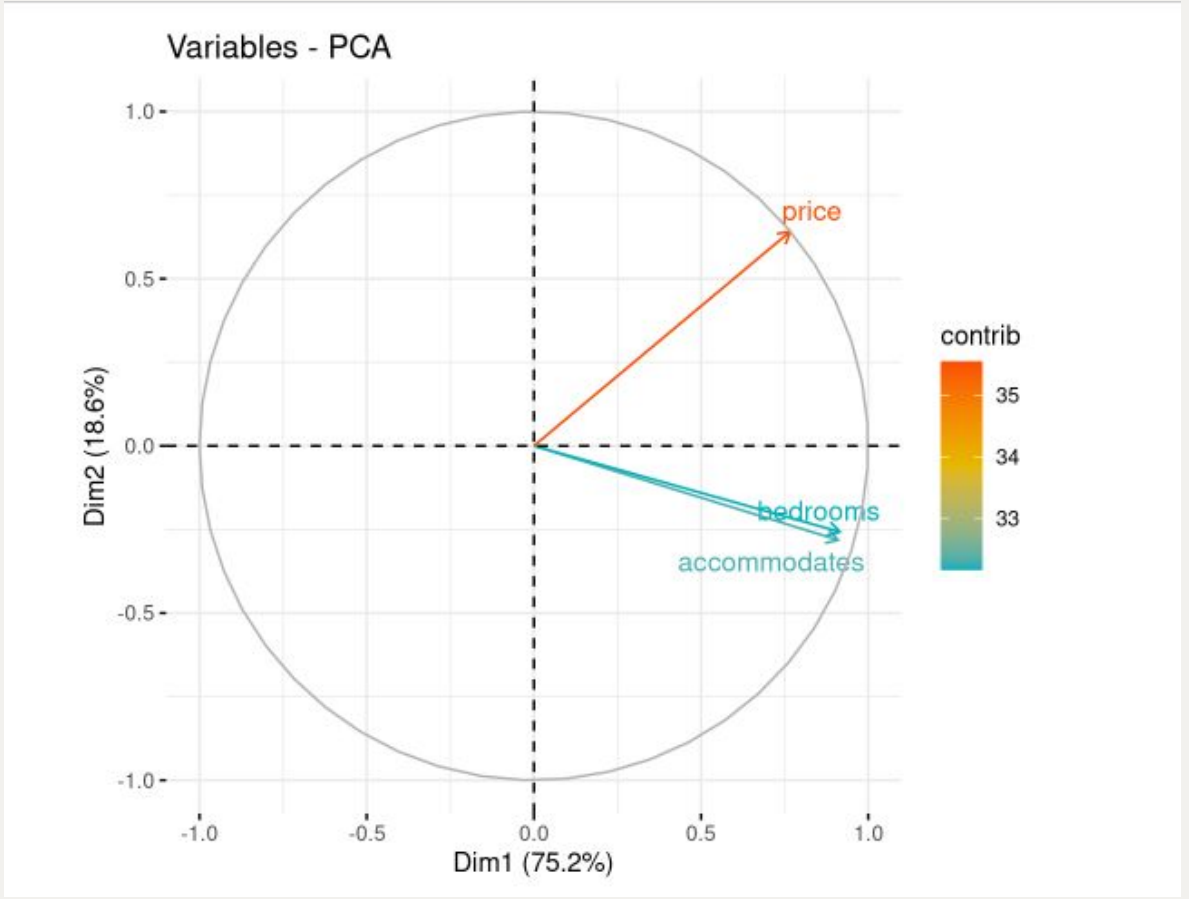
- Airbnb as a platform benefits both travelers and hosts
- Overall satisfaction rating in Asheville
- Prices of a listings based on distance to center city

02 Data Cleaning

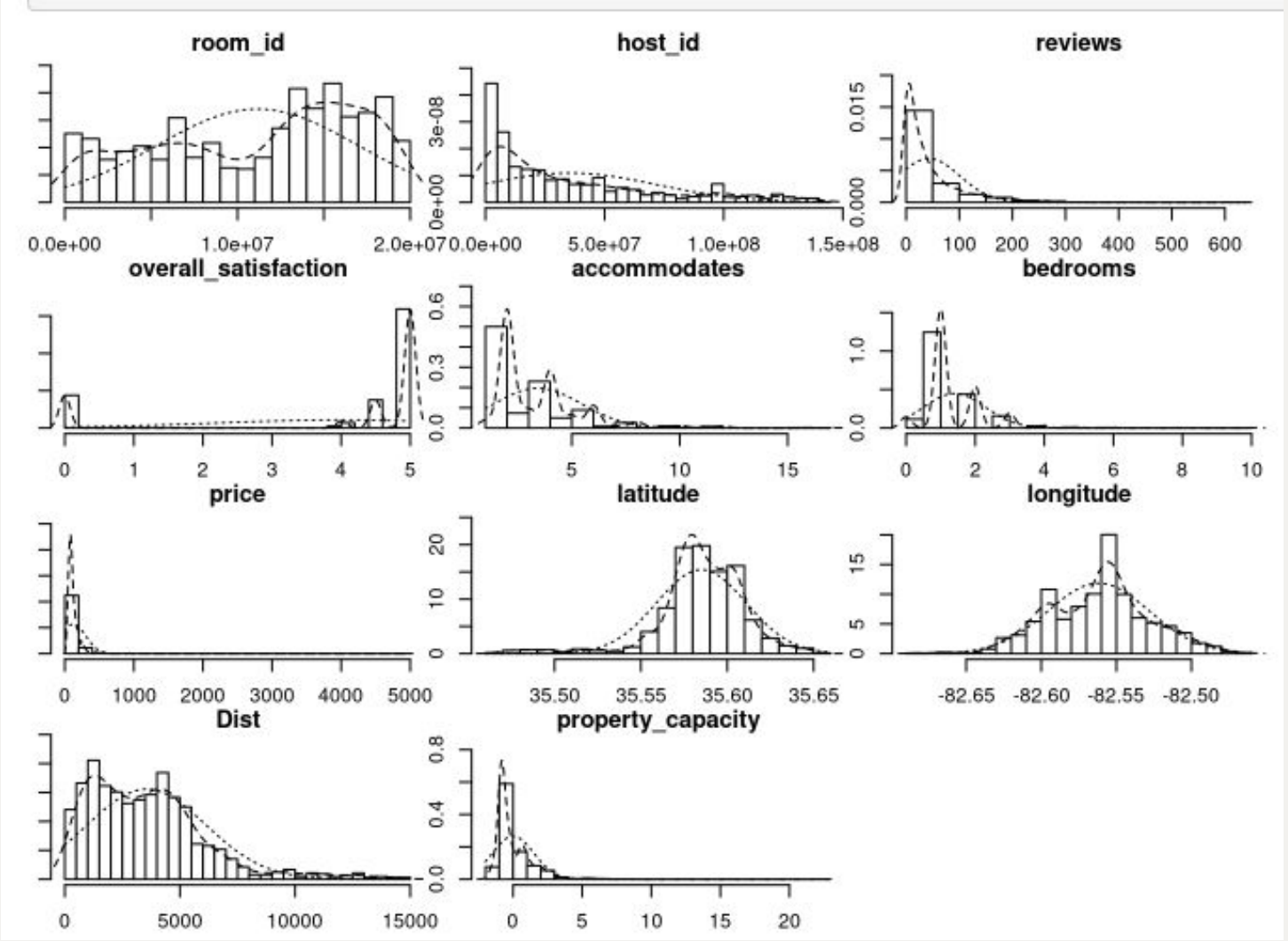
- Features are chosen only if they are informative and likely to be correlated with the listing price
- Removed features that may appear to be noise, have duplicate features or where majority of data is null
- New variable, ‘Dist’, which uses latitude and longitude variables to calculate the distance of each listing from a central point in downtown Asheville.
- We created a dummy variable that captures if the customer was satisfied with the room or not.
- We transformed the categorical variables into dummies and renamed them

Data Cleaning Cont. - PCA





Numeric Variables Exploration



03 Analysis Methods & Objectives

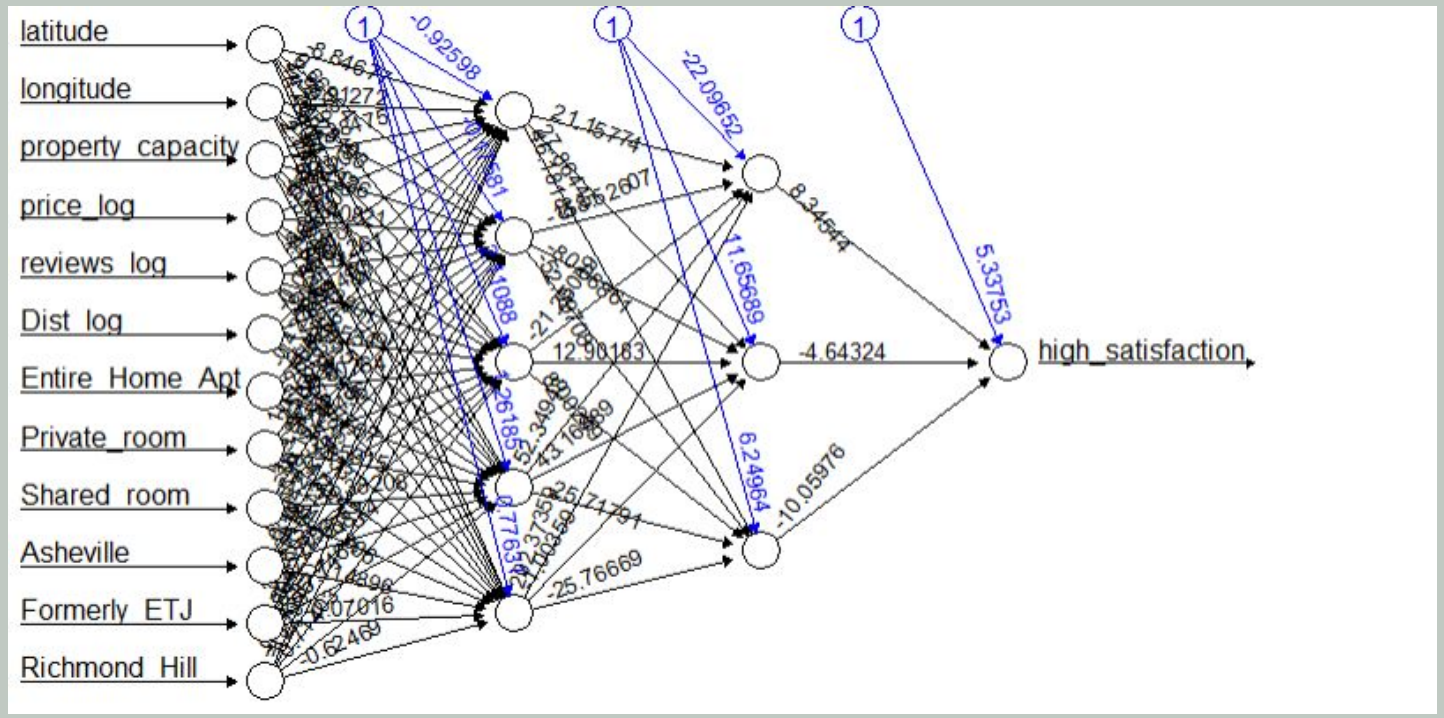
- Key objective was to predict Airbnb listing price and the satisfaction of a listing based on various modeling techniques

- The models used in this project were linear regression, text mining, neural network, random forest, KNN, XGboost and decision trees

High Satisfaction: Neural Network Performance

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0  70  11
         1   7 304

               Accuracy : 0.9541
                 95% CI : (0.9284, 0.9726)
    No Information Rate : 0.8036
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8573

 Mcnemar's Test P-Value : 0.4795

            Sensitivity : 0.9651
            Specificity : 0.9091
         Pos Pred Value : 0.9775
         Neg Pred Value : 0.8642
             Prevalence : 0.8036
         Detection Rate : 0.7755
   Detection Prevalence : 0.7934
      Balanced Accuracy : 0.9371

       'Positive' Class : 1
```

- Neural Network was the worst out of the two models at predicting the High Satisfaction variable
- NN Model had an accuracy of 95.41% and a Kappa of .8573

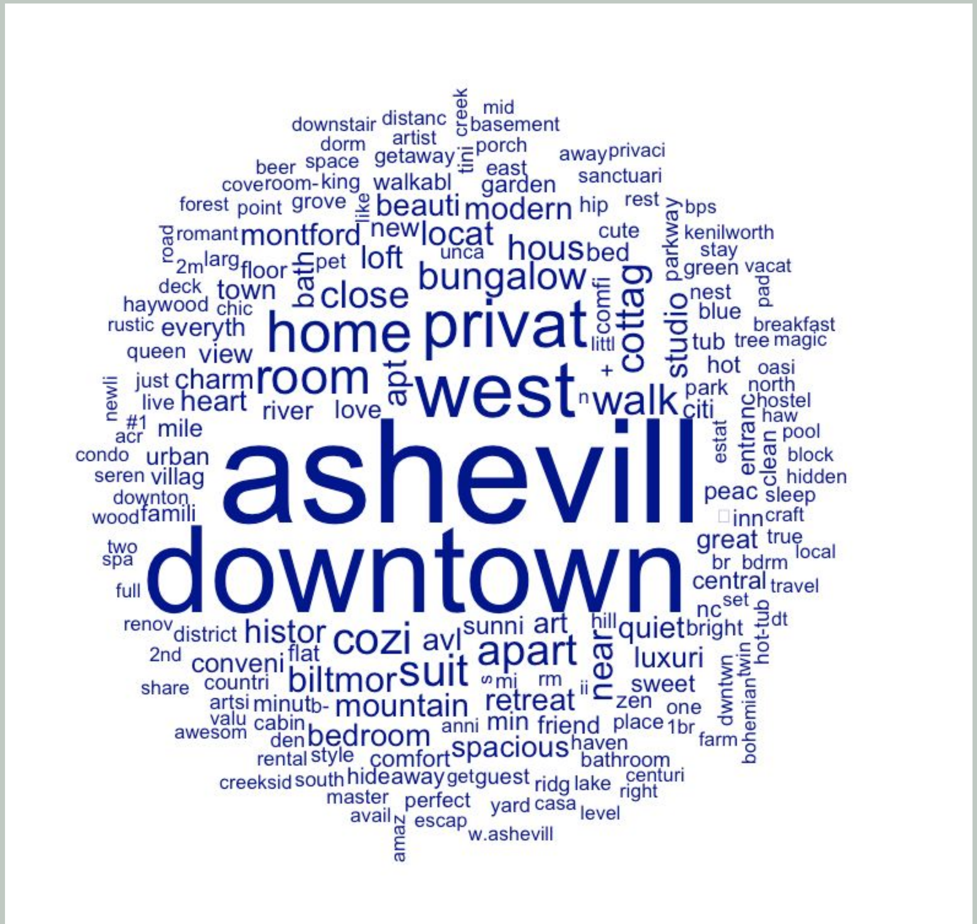


High Satisfaction: Random Forest Performance

```
Random Forest

915 samples
 12 predictor
  2 classes: '0', '1'

No pre-processing
Resampling: Bootstrapped (25 reps)
Summary of sample sizes: 915, 915, 915, 915, 915, 915, ...
Resampling results across tuning parameters:

  mtry  Accuracy   Kappa
   2    0.9652540  0.8861652
   7    0.9648751  0.8847092
  12    0.9626374  0.8764794

Accuracy was used to select the optimal model using the largest value.
The final value used for the model was mtry = 2.
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0  63   9
         1   0 320

               Accuracy : 0.977
                 95% CI : (0.9569, 0.9894)
    No Information Rate : 0.8393
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9195

 Mcnemar's Test P-Value : 0.007661

            Sensitivity : 1.0000
            Specificity : 0.9726
         Pos Pred Value : 0.8750
         Neg Pred Value : 1.0000
             Prevalence : 0.1607
         Detection Rate : 0.1607
   Detection Prevalence : 0.1837
      Balanced Accuracy : 0.9863

       'Positive' Class : 0
```

- Best performing model out of the models we tried for predicting High Satisfaction
- Random Forest Model has an accuracy of 97.7% and a Kappa of .9195
- The RF Model is a more accurate model when compared to the Neural Network Model and a Regular Decision Tree Model

Text Mining - Wordcloud



ashevill	downtown	west	privat	home	room	cozi	suit
1022	829	483	400	325	264	220	217
apart	walk	cottag	bungalow	close	biltmor	near	hous
208	201	183	173	164	147	144	143
locat	apt	histor	mountain	studio	loft	modern	avl
133	129	124	118	117	115	112	107
bedroom	retreat	beauti	montford	quiet	bath	spacious	art
107	106	105	102	101	97	96	93
heart	luxuri	charm	town	view	conveni	new	great
92	90	85	82	74	73	72	72
citi	bed	river	min	everyth	sunni	love	friend
69	67	66	66	64	61	61	56
peac	mile						
56	56						

Models that predict price

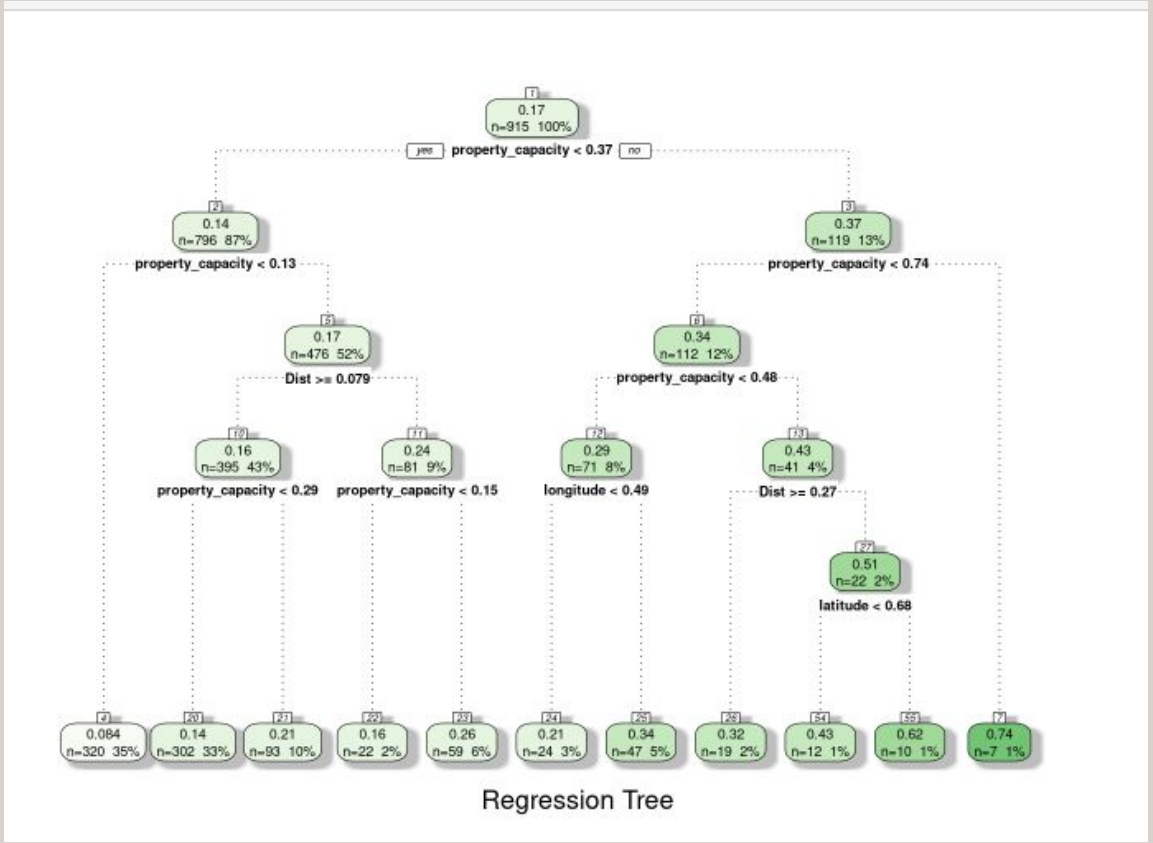
Linear Regression

- Linear Regression was the most interpretable model
- Linear Regression provides insight about how the price changes when we increase in one unit one of the explanatory variables holding the other variables constant
- Distance and property capacity are the most important predictor variables
- High satisfaction and Entire Home / Apt are also important price predictors

```
Call:
glm(formula = price ~ reviews + high_satisfaction + latitude +
    longitude + property_capacity + Dist + Richmond_Hill + Formerly_ETJ +
    Private_room + Entire_Home_Apt, data = Asheville.train)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.27326  -0.04243  -0.00482   0.03283   0.48154

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       -0.019657   0.044347  -0.443  0.65769
reviews           -0.067057   0.030891  -2.171  0.03021 *
high_satisfaction -0.024020   0.007703  -3.118  0.00188 **
latitude          -0.006819   0.025827  -0.264  0.79182
longitude          0.058487   0.017846   3.277  0.00109 **
property_capacity  0.591623   0.022684  26.081  < 2e-16 ***
Dist              -0.134879   0.021423  -6.296 4.76e-10 ***
Richmond_Hill     -0.023934   0.082562  -0.290  0.77196
Formerly_ETJ       0.014467   0.011170   1.295  0.19558
Private_room       0.064463   0.037130   1.736  0.08288 .
Entire_Home_Apt    0.105628   0.037115   2.846  0.00453 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for gaussian family taken to be 0.00676425)

    Null deviance: 15.0368  on 914  degrees of freedom
Residual deviance:  6.1149  on 904  degrees of freedom
AIC: -1961.8
```

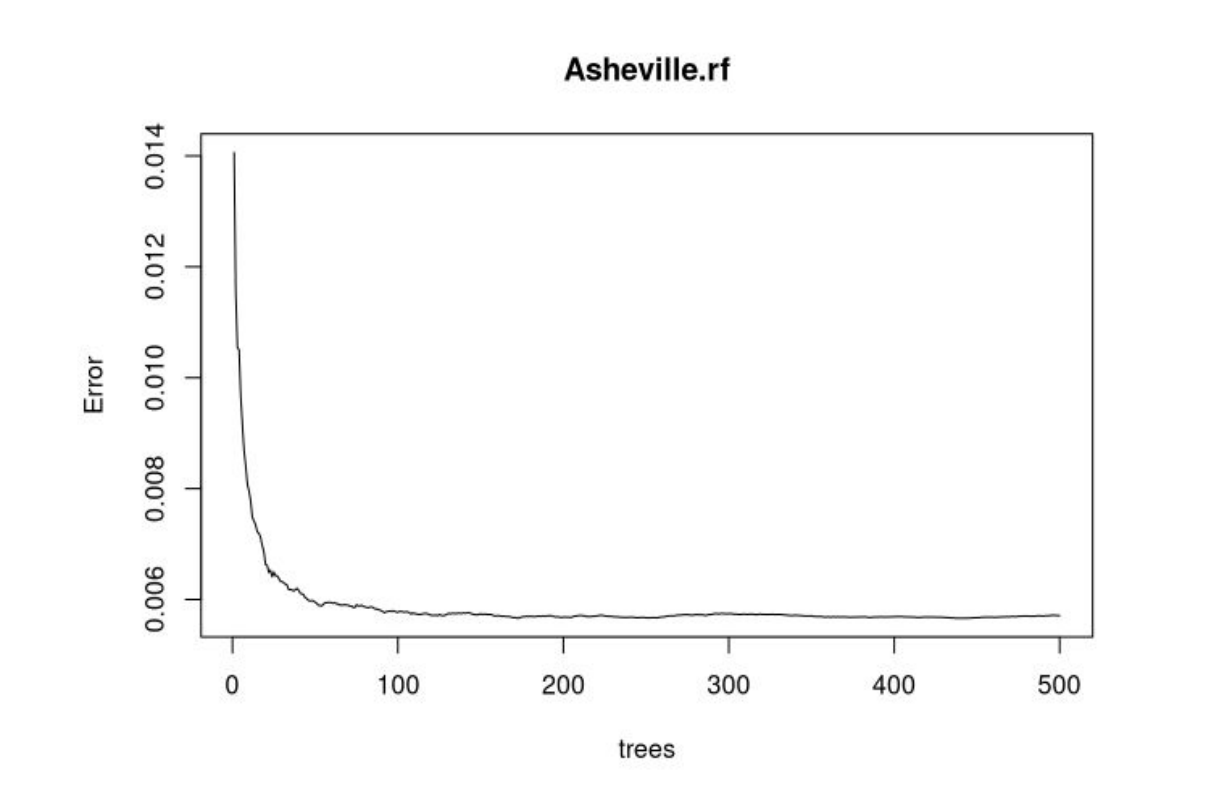
Decision Tree

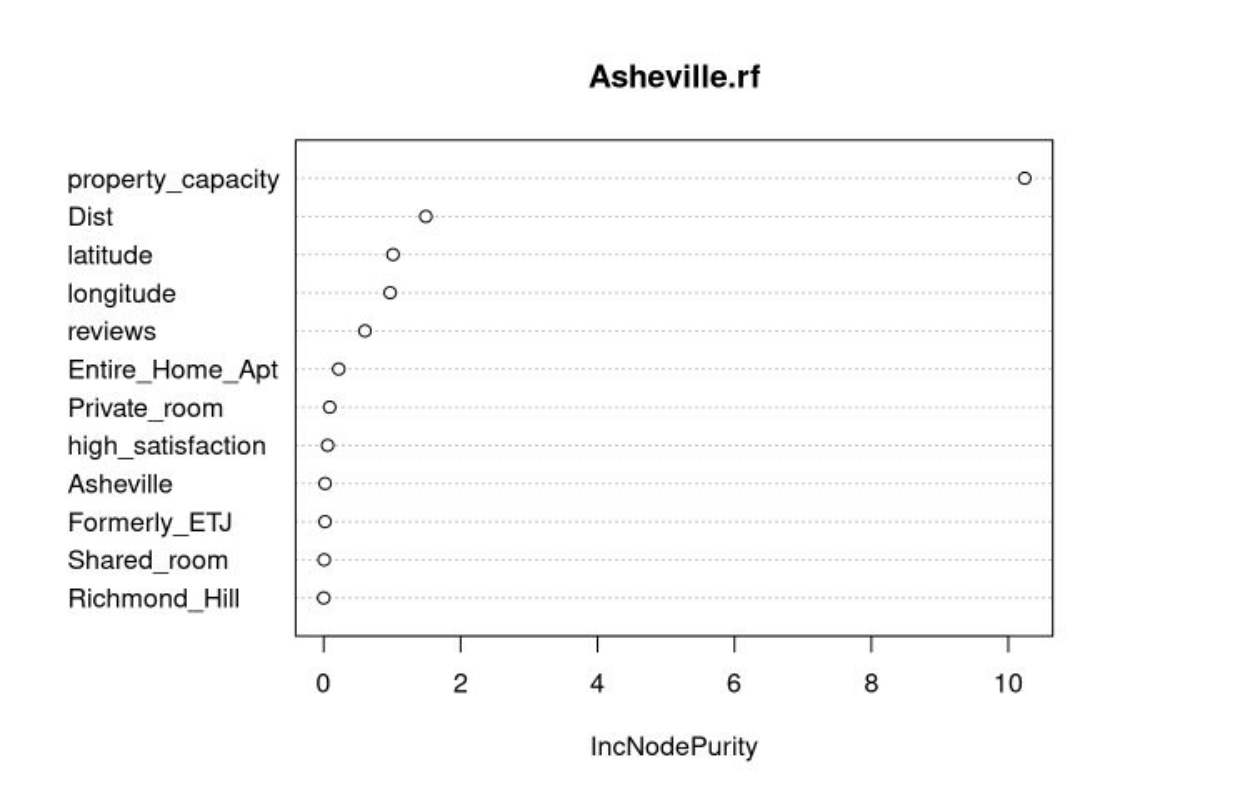
- Decision tree was also easy to interpret
- The plot describes the rules the algorithm followed to predict the price
- Distance and property capacity are the most important predictor variables followed by latitude and longitude



Regression Tree

Random Forest





- The random forest also finds that distance, property capacity, latitude and longitude are the most important predictor variables
- The MSE drops significantly after 100 trees

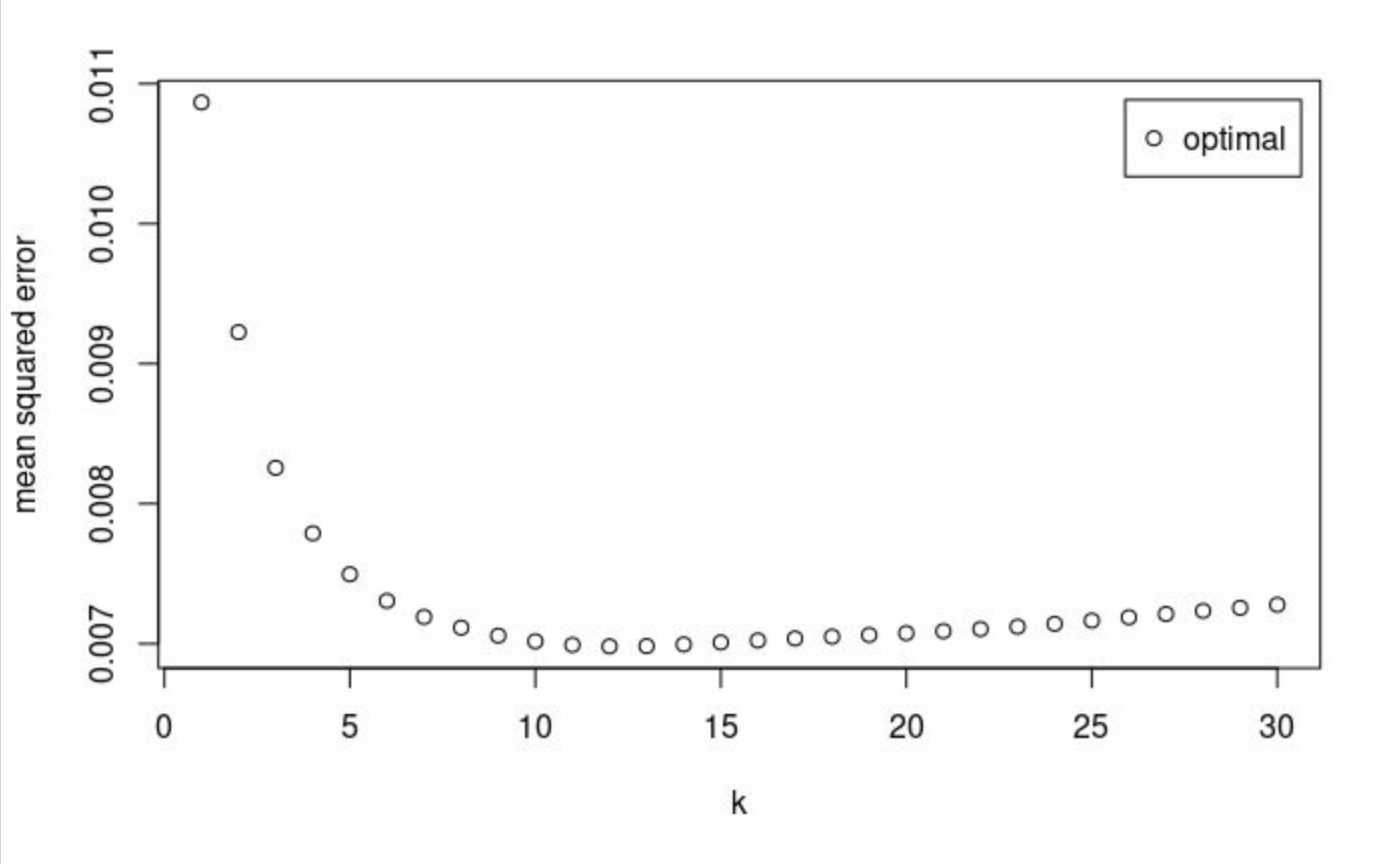
XGBOOST

- XGBoost uses decision trees for its base learners
- This process slowly learns from data and tries to improve its prediction at subsequent iterations
- The graph in the left shows the parameters used for the tree booster

```r
xgb_grid <- expand.grid(nrounds = 50,
                        max_depth = 5,
                        eta = 0.1,
                        gamma=0,
                        colsample_bytree = 0.9,
                        min_child_weight = 1,
                        subsample = 1
)
# Set up data frame with tuning parameters:
# nrounds – Number of Boosting Iterations
# max_depth – Max Tree Depth
# colsample_bytree – Subsample Ratio of Columns
# eta – Shrinkage
# gamma – Minimum Loss Reduction
# min_child_weight – Minimum Sum of Instance Weight
# subsample – Subsample Percentage
```

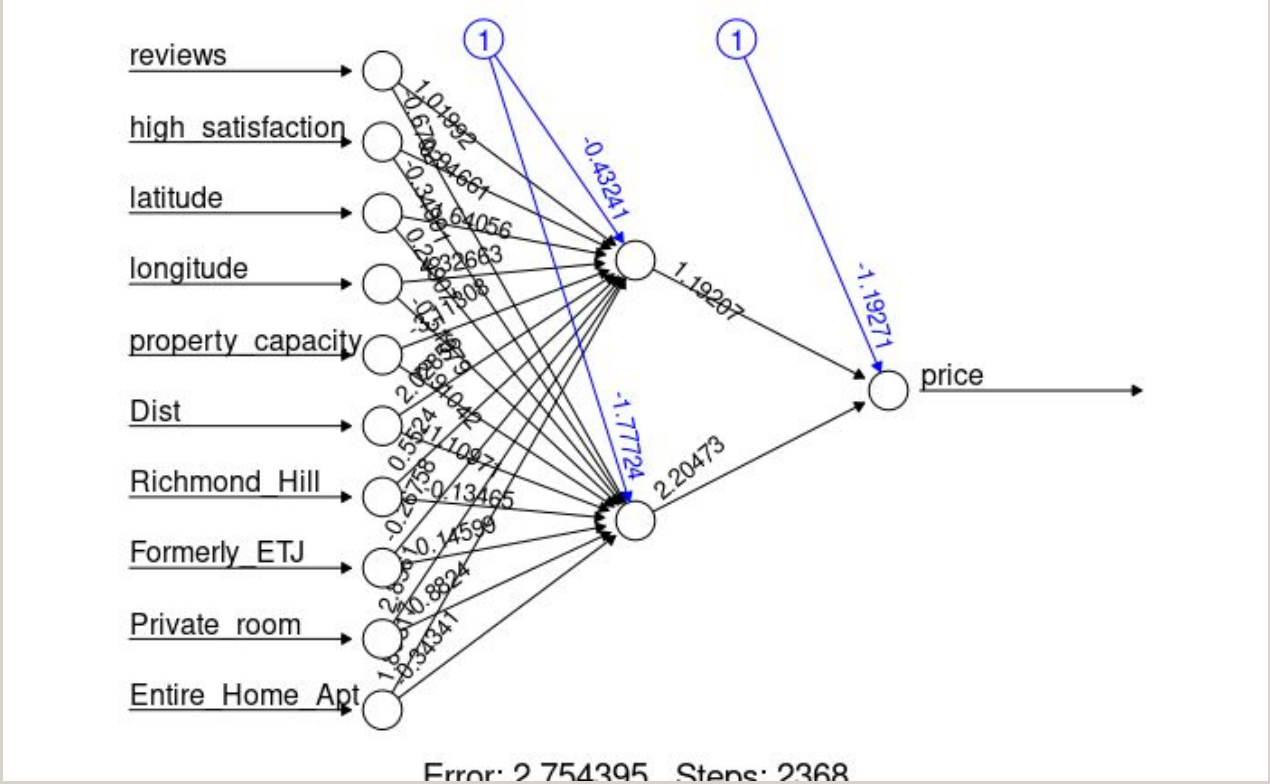
KNN

- KNN is an algorithm that is useful for matching a point with its closest neighbours in a multidimensional space
- When predicting a continuous target variable, the prediction is made to some values of the K nearest neighbors
- To find the optimal K, we choose the model that minimizes the MSE for all k between 1-30
- The minimum MSE is achieved when k = 12



Neural Network

- NN tends to overfit to the training dataset
- We tried various numbers of hidden layers and units
- The model which performed the best has one hidden layer with two units. The second best model had four hidden layers with 4,3,2,1 units
- We found that simpler neural architectures generalised better to the test data



04 Model Performance

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

Models without text features (price)		
	RSME	R-squared
Linear regression	0.08461779	0.5855216
KNN	0.08451375	0.5785441
Decision Tree	**0.09036185**	**0.5181991**
Xgboost	0.07422601	0.6975985
Random Forest	**0.07158851**	**0.7043414**
Neural network	0.08087062	0.6140964

Models with text features (price)		
	RSME	R-squared
Linear regression	0.08141777	0.5966286
KNN	**0.1013182**	**0.3942798**
Decision Tree	0.0894111	0.5282844
Xgboost	0.07777471	0.6430774
Random Forest	**0.07579417**	**0.6905551**
Neural network	0.08241416	0.5992247

05 Future Work

- Test the models' performance with other city datasets that have more features
- Keep tuning the algorithms to find the best parameters
- Test which text features improve performance more (word counts, unigrams, bigrams or unigram+bigrams)
- Survey property owners to find out how they set prices
- Survey Airbnb clients to find out how they rate a property

Questions?