**3.nl KNN**

We apply KNN method to predict price. The most primitive price prediction algorithm of Airbnb is to use the idea of KNN, which predicts the most appropriate price of a new listing from a set of geographically close listings. Since KNN-regression users Euclidean distance to define the neighbor, only quantitative predictors will be used. We add in all feasible quantitative predictors by 3 steps, as shown in Table XX. The first group starts with geographical information, as the primitive Airbnb algorithm, and then add in other predictors. Within each group, we conduct cross-validation to determine the best number of neighbors—k.

Table 1 Training and testing RMSE of KNN regression

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Group 1** | |  | **Group 2** | |  | **Group 3** | |
|  | Longitude, latitude | |  | Group 1 + beds, bedrooms, accommodates, bathrooms, amenities | |  | Group 2 + review scores rating, number of reviews, host response rate | |
| **k** | Train RMSE | Test RMSE |  | Train RMSE | Test RMSE |  | Train RMSE | Test RMSE |
| 1 | 0.00 | 1.02 |  | 0.00 | 1.02 |  | 0.00 | 1.02 |
| 5 | 0.54 | 0.86 |  | 0.42 | 0.92 |  | 0.44 | 0.91 |
| 10 | 0.58 | 0.83 |  | 0.46 | 0.90 |  | 0.47 | 0.89 |
| 25 | 0.60 | 0.82 |  | 0.49 | 0.88 |  | 0.50 | 0.87 |
| 50 | 0.61 | 0.81 |  | 0.50 | 0.87 |  | 0.52 | 0.86 |
| 250 | 0.63 | 0.79 |  | 0.53 | 0.84 |  | 0.54 | 0.84 |
| 500 | 0.64 | 0.78 |  | 0.55 | 0.83 |  | 0.56 | 0.82 |
| 840 | 0.65 | 0.77 |  | 0.56 | 0.81 |  | 0.57 | 0.80 |
| 1,000 | 0.65 | 0.77 |  | 0.57 | 0.80 |  | 0.58 | 0.80 |
| 3,000 | 0.67 | 0.75 |  | 0.61 | 0.77 |  | 0.62 | 0.77 |
| 5,000 | 0.69 | 0.74 |  | 0.64 | 0.76 |  | 0.65 | 0.75 |
| 10,000 | 0.72 | 0.72 |  | 0.70 | 0.73 |  | 0.70 | 0.73 |

Note: All predictors except for longitude and latitude are scaled before training model. Data are randomly divided into training and testing datasets. The best k is searched from a series from 1 to 10,000. Note that in training dataset, the numbers of listings in each city are in the range of 1,264 (Boston) to 11634 (New York City).

**3.nb Regression Tree**

We apply regression tree method to predict prices of Airbnb listings. We grow a tree using all the feasible predictors in training data and conduct a 10-fold cross-validation to prune the tree. The initial complexity parameter is set to 0.0002. As shown in Figure XX, the unpruned tree has 235 splits. The best prune tree with minimized CV-RMSE has 166 splits.

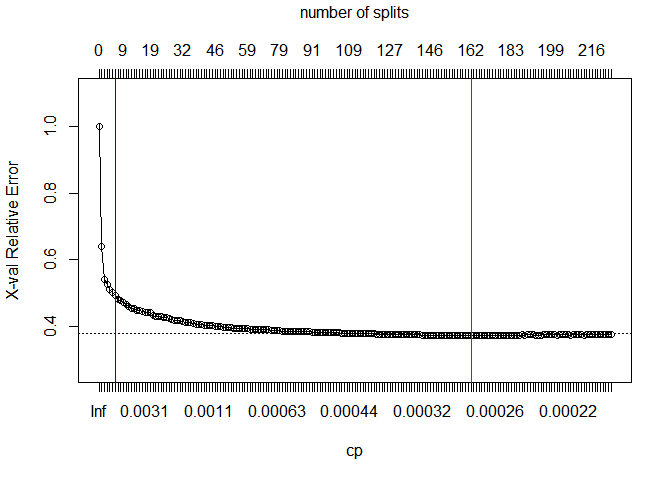


Figure 1 The cross-validation RMSE of regression tree

Note: The lower x-axis is the complexity parameter associated with each pruned tree. The upper x-axis is the number of splits. The red line indicates the best pruned tree with minimized cross-validation RMSE. The blue line indicates a smaller tree with 7 splits. The horizontal line is the 1SE above the minimum of the curve.

In Figure XX we report the graph of the best tree. Since the best tree is too large to interpret, we also report a smaller tree with 7 splits, which is the trunk of the best tree. The most important factor of price is room type. Entire houses/apartments usually have higher prices than private rooms, and a private rooms have higher price than shared rooms. Location also affect price largely. From the graph, listings in Chicago, LA and San Francisco are more likely claiming a higher price. The training and testing RMSE of two models are reported in Table XX.

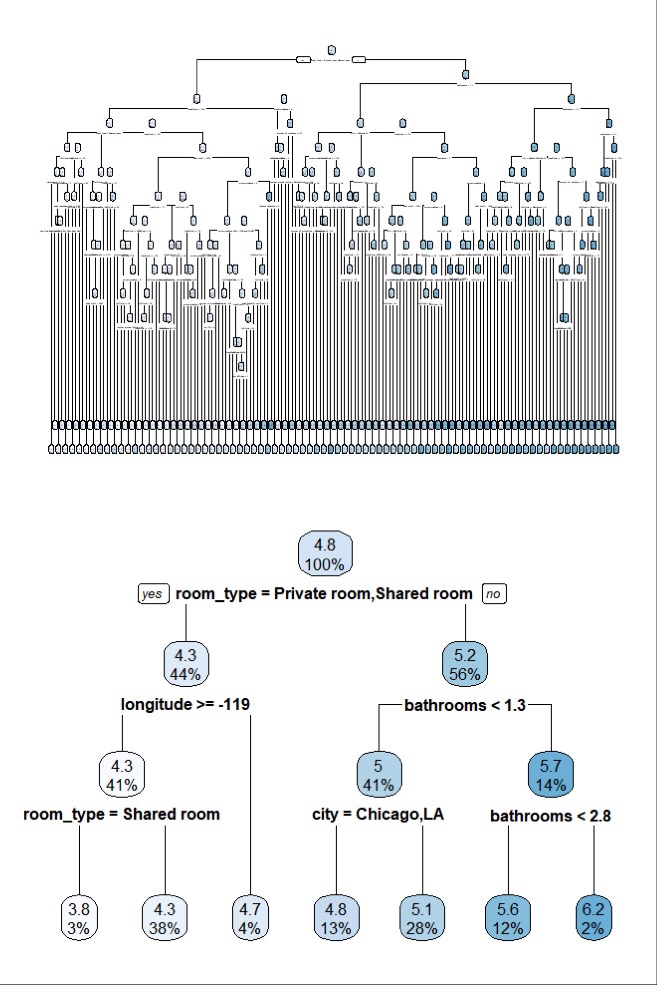


Figure 2 Pruned trees

Note: The upper graph is the best pruned tree with lowest CV-RMSE. The lower graph shows the trunk of the best tree. Longitude -119 is roughly the eastern edge of the California state, bound out SF and LA in the dataset.

Table 2 Training and testing RMSE of pruned tress

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | Best Tree |  | 7-split Tree |  |
| **Training RMSE** |  | 0.4117 |  | 0.5026 |  |
| **Testing RMSE** |  | 0.4456 |  | 0.5045 |  |