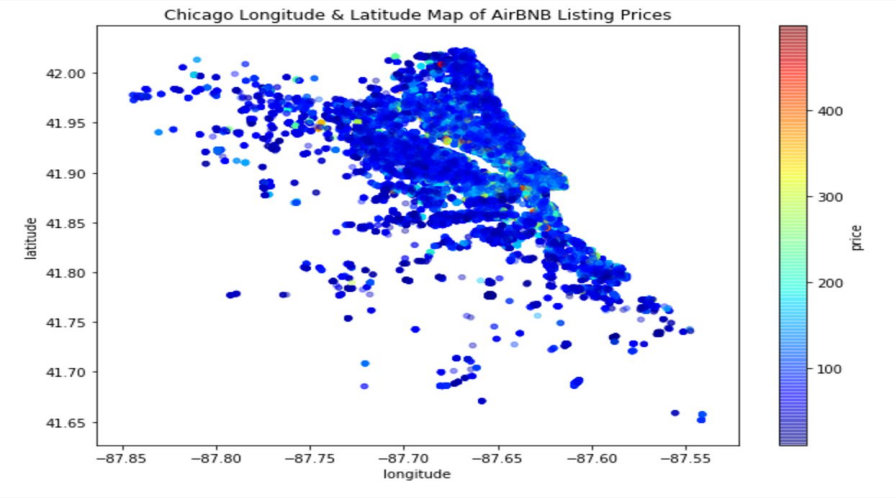

# Airbnb Price Point

## What's YOUR home worth?

CFrensko

# Overview:

- When visiting a city for the first time or renting out your home it can be a struggle figuring out if you are getting a fair price for the AirBNB rental or whether you priced your home appropriately to make a profit.
- The purpose of this project is to use machine learning to calculate a fair approximate value for a home to assist renters and hosts using AirBNB.



# Data Collection:

- The data used for this project was pulled from AirBNB's website by Tom Slee (http://tomslee.net/airbnb-data-collection-get-the-data).
- The dates of the data are from 26 randomly selected days over the past five years.
- I downloaded the separate csv files for each date using glob and joined them together to form one dataframe.
- The data used totals 159,181 rows for the city of Chicago.

**Chicago**

Survey dates: 2013-12-23 (2095 listings), 2014-05-13 (1979 listings), 2015-02-18 (3196 listings), 2015-10-03 (4989 listings), 2015-10-23 (5513 listings), 2015-11-26 (5843 listings), 2015-12-15 (6000 listings), 2016-01-19 (5954 listings), 2016-02-21 (5962 listings), 2016-03-21 (5999 listings), 2016-04-17 (6140 listings), 2016-05-21 (6100 listings), 2016-06-20 (6270 listings), 2016-07-18 (6582 listings), 2016-08-21 (6707 listings), 2016-09-18 (6799 listings), 2016-10-21 (7230 listings), 2016-11-24 (8343 listings), 2016-12-24 (8520 listings), 2017-01-14 (8207 listings), 2017-02-17 (8239 listings), 2017-03-13 (8088 listings), 2017-04-09 (7922 listings), 2017-05-06 (5080 listings), 2017-06-11 (5613 listings), 2017-07-11 (5811 listings)

# Data Wrangling:

- I began by performing exploratory data analysis.
- Columns missing greater than 90% of  data, columns specific to AirBNB's platform, & null values were removed.
- Categorical columns listed as strings were changed to integers with dummy variables.
- Rows with less than 3 reviews were removed & null values under minimum stay were changed to a median value.
- Outlier listings priced above $500 & outliers in the minimum stay column were removed.
- To avoid multicollinearity the accommodates variable was eliminated.

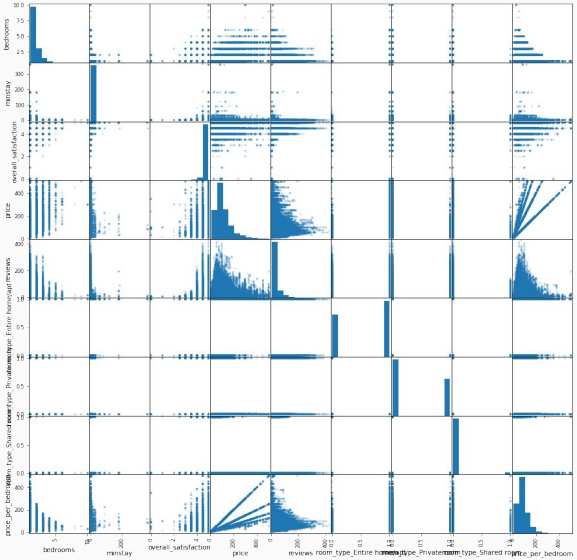
**Replace room type strings with dummy variable for comparison**

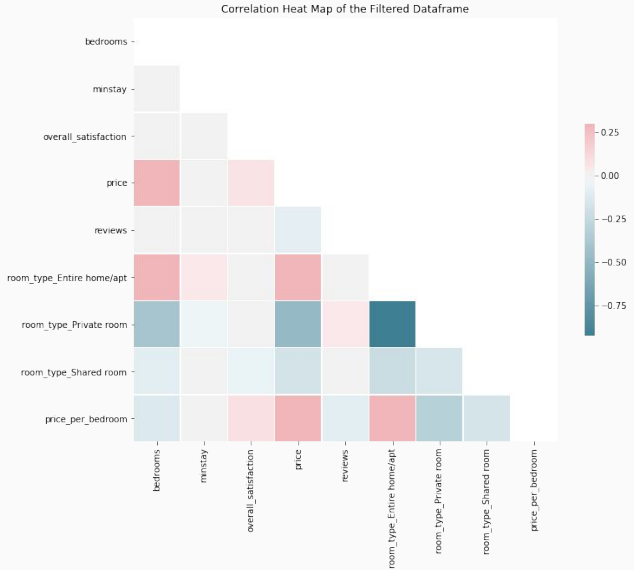
```
dummies = pd.get_dummies(result['room_type']).rename(columns=lambda x: 'room_type_' + str(x))
filtered_result = pd.concat([result, dummies], axis=1)
filtered_result = filtered_result.drop(['room_type'], axis=1)
filtered_result.head()
```

| | accommodates | bedrooms | minstay | neighborhood | overall_satisfaction | price | reviews | room_type_Entire home/apt | room_type_Private room | room_type_Shared room |
|---|---|---|---|---|---|---|---|---|---|---|
| **1** | 8.0 | 3.0 | 2.0 | Lawndale | 5.0 | 98.0 | 186 | 1 | 0 | 0 |
| **3** | 14.0 | 4.0 | 4.0 | Wrightwood Neighbors | 4.5 | 296.0 | 46 | 1 | 0 | 0 |
| **7** | 8.0 | 4.0 | 3.0 | Lake View East | 4.5 | 301.0 | 2 | 1 | 0 | 0 |
| **14** | 10.0 | 4.0 | 3.0 | Bucktown | 4.0 | 526.0 | 1 | 1 | 0 | 0 |
| **19** | 2.0 | 1.0 | 3.0 | Old Town | 5.0 | 108.0 | 50 | 1 | 0 | 0 |

# Exploratory Data Analysis

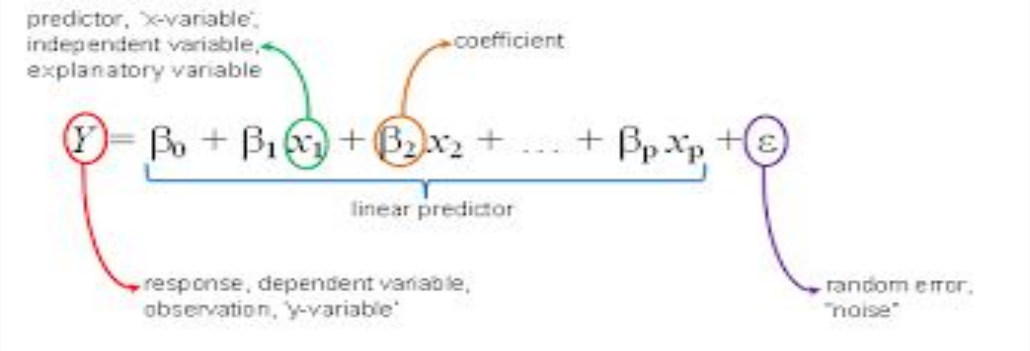
- After the data was cleaned I created a scatterplot matrix and heatmap (without the neighborhood data) to get an initial overview of the data set.



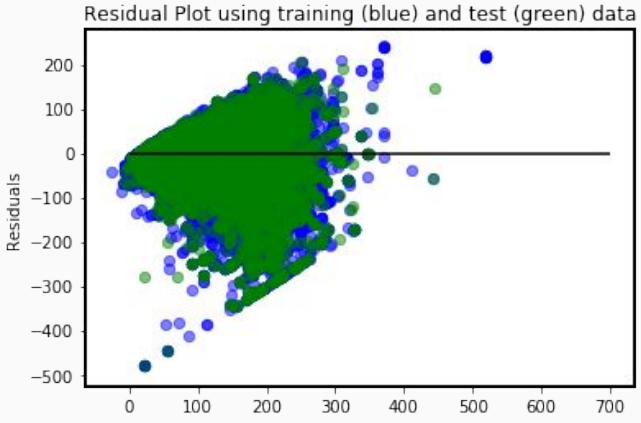
# Linear Regression Model:

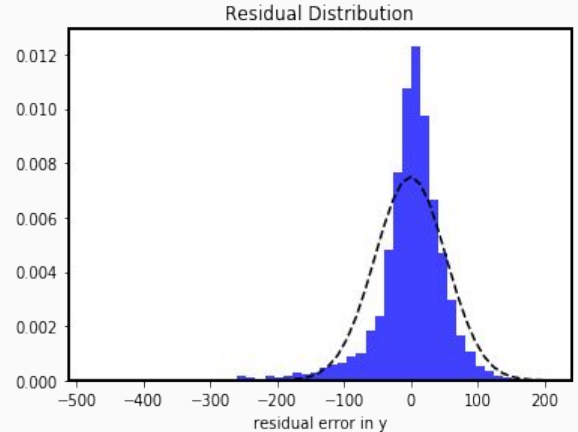
- The clean data set was split into a 80/20 train/test set and fit to a linear regression model to predict price.
- 52.18% of the variability in Y (the price) can be explained using X (the 198 other coefficients).
- Only 40 percent of the test set values were within 20 dollars of the price.
- The root mean squared error (RMSE) value was 53.2029
- Which means our model predicts the value of every AirBNB home in the test set within $53.20 of the real price.
- With a mean price was $113 with a standard deviation of $76 these results are not that exciting yet...

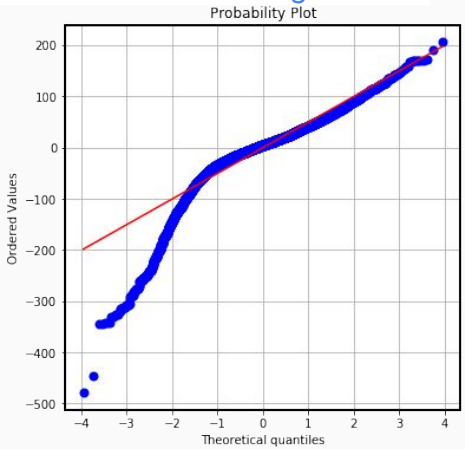


# Linear Regression Residual Charts:

- Next I calculated and plotted the residuals of both the train and test sets to observe the estimates of experimental error obtained by subtracting the observed responses from the predicted responses.
- Residual plots are a good way to visualize the errors in data.
- Accurate residual plots should be randomly scattered around line zero.
- Accurate residual histograms should have a normal Gaussian distribution.
- An accurate probability plot should show normality of the distribution of the residual errors following the line.

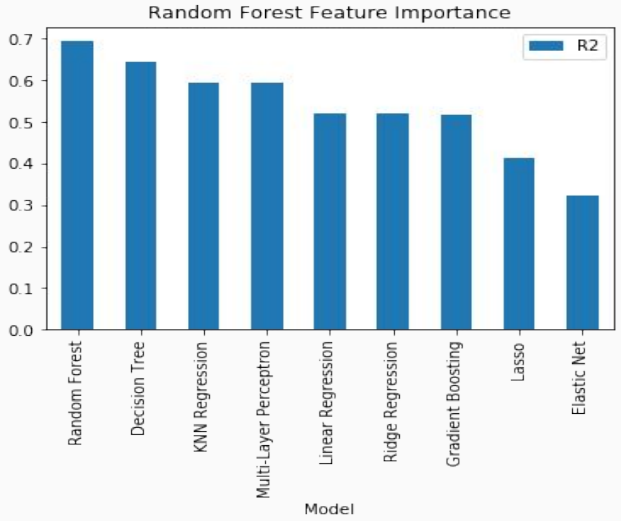


# Supervised Learning:

- Next I applied a variety of supervised regression models.
- I used the same test/train split and fit the training data to predict price.

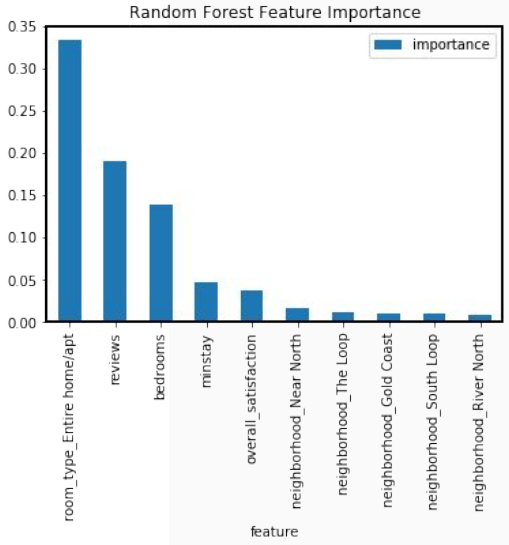



| Model | R2 | RMSE | Bias |
|---|---|---|---|
| **Random Forest** | 0.694443 | 42.529798 | -0.211461 |
| **Decision Tree** | 0.643155 | 45.960707 | -0.601326 |
| **KNN Regression** | 0.594959 | 48.966190 | 0.495579 |
| **Multi-Layer Perceptron** | 0.592619 | 49.107463 | -0.234844 |
| **Linear Regression** | 0.521836 | 53.202906 | -0.561742 |
| **Ridge Regression** | 0.521752 | 53.207590 | -0.562194 |
| **Gradient Boosting** | 0.517556 | 53.440502 | -0.575509 |
| **Lasso** | 0.414591 | 58.867637 | -0.370890 |
| **Elastic Net** | 0.321566 | 63.372496 | -0.700197 |

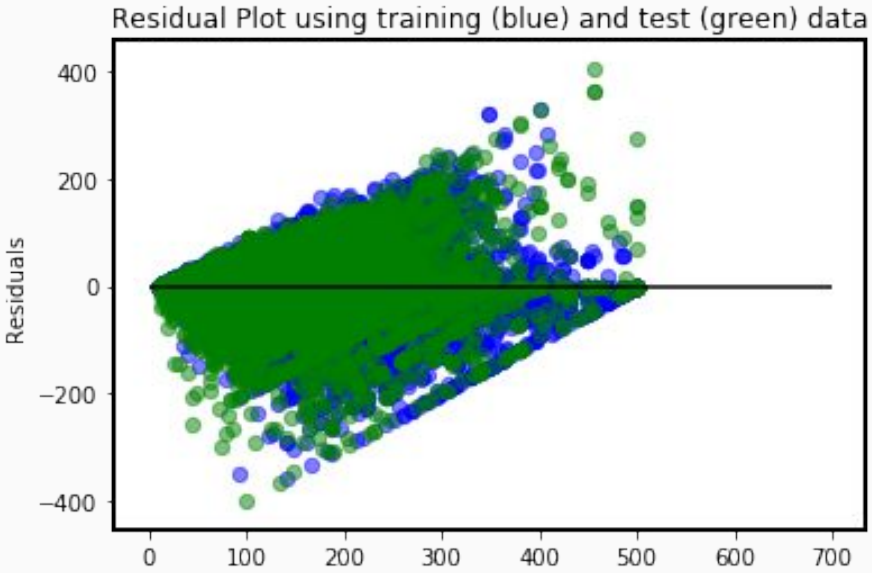
# Random Forest Feature Analysis:

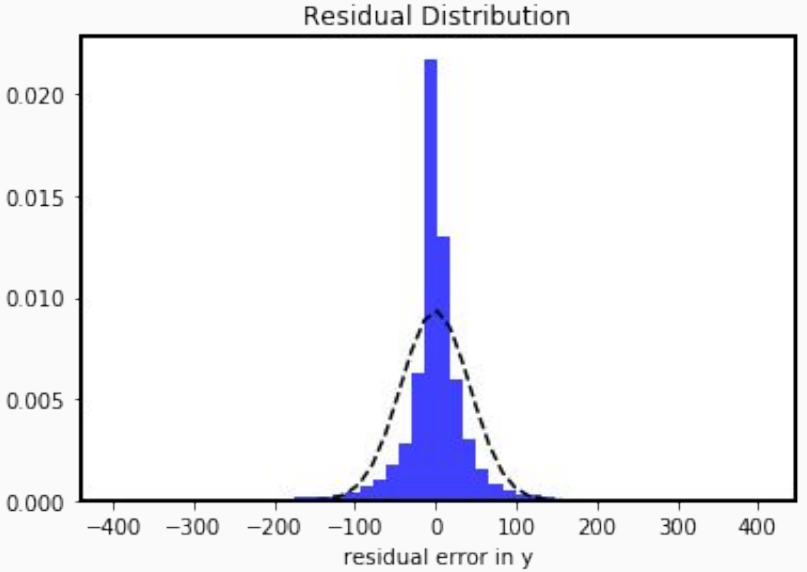
- The top weighted features for determining price using the random forest model (out of 196 features):

| feature | importance |
|---|---|
| room_type_Entire home/apt | 0.3329 |
| reviews | 0.1910 |
| bedrooms | 0.1393 |
| minstay | 0.0465 |
| overall_satisfaction | 0.0378 |
| neighborhood_Near North | 0.0175 |
| neighborhood_The Loop | 0.0124 |
| neighborhood_Gold Coast | 0.0101 |
| neighborhood_South Loop | 0.0099 |
| neighborhood_River North | 0.0095 |
| neighborhood_Old Town | 0.0087 |
| neighborhood_Near East Side | 0.0084 |
| neighborhood_Lake View East | 0.0066 |
| neighborhood_Logan Square | 0.0064 |
| neighborhood_Lake View | 0.0060 |
| neighborhood_Old Town Triangle | 0.0053 |
| neighborhood_Roscoe Village | 0.0053 |
| neighborhood_Lincoln Park | 0.0052 |
| neighborhood_Wicker Park | 0.0049 |
| neighborhood_Humboldt Park | 0.0047 |
| neighborhood_Fulton River District | 0.0046 |
| neighborhood_Bucktown | 0.0045 |
| neighborhood_West Town | 0.0045 |
| neighborhood_Wrigleyville | 0.0045 |
| neighborhood_Sheffield Neighbors | 0.0044 |
| neighborhood_River West | 0.0043 |
| neighborhood_Near West Side | 0.0042 |
| neighborhood_East Ukrainian Village | 0.0039 |
| neighborhood_Uptown | 0.0038 |
| neighborhood_Dearborn Park | 0.0038 |



# Random Forest Residual Analysis:



# Tuning Hyperparameters for Random Forest Model:

- A model may perform well on the training set but if it is overfit it will be useless in a real application.
- Hyperparameter optimization accounts for overfitting through cross validation.
- I used random search of parameters, using 3 fold cross validation to fit each of 100 candidates totalling 300 fits.
- The best parameter fit for the model is shown below:

```
rf_random.best_params_
{'bootstrap': True,
 'max_depth': 100,
 'max_features': 'sqrt',
 'min_samples_leaf': 1,
 'min_samples_split': 5,
 'n_estimators': 500}
```

| Model | R2 | RMSE | Bias |
|---|---|---|---|
| Random Forest Hyperparameters Tuned | 0.716490 | 40.966734 | -0.199421 |
| Random Forest | 0.694443 | 42.529798 | -0.211461 |

# Conclusions

Our results may be summarized as follows:

- Using the random forest model to predict prices on the test set gave us a R2 value of 0.6952.
- This indicates 69.52% of the variability in price can be explained using this model.
- With hyperparameters tuned the model gave us a 72% explanation of the variability of the test set.
- This is a pretty good indicator of price.
- However there is still much room for improvement!

# Future Work

Since the focus of this work is to not only predict the price for Airbnb rentals but also to understand the fluctuation in the variables more research is needed. In particular:

- Gather a larger more robust set of data.
- Perform a  time series analysis.
- Include amenities and facilities in the next analysis.
- Take data from rental costs of homes in Chicago to improve model's accuracy.
- Create an application to generate key price ranges.
- Scale the price model for apartments on longer term leases.

# Client Recommendations

Once these improvement have been made, the model can be used to generate an even more accurate prediction of prices for AirBnb home rentals in Chicago.

- Before you book that trip to a new city check out the price ranges on the application.
- If you are hosting you will easily structure the pricing of your unit with fair market values listed.
- Also for long term leases, pull out the application to navigate the negotiation for a more affordable price!

# Thank you!

- Special thanks to Springboard & my mentor AJ Sanchez!

# Title: Airbnb Price Point:

# What’s YOUR home worth?


**Created by: Caroline Frensko**

**cfrensko@gmail.com**