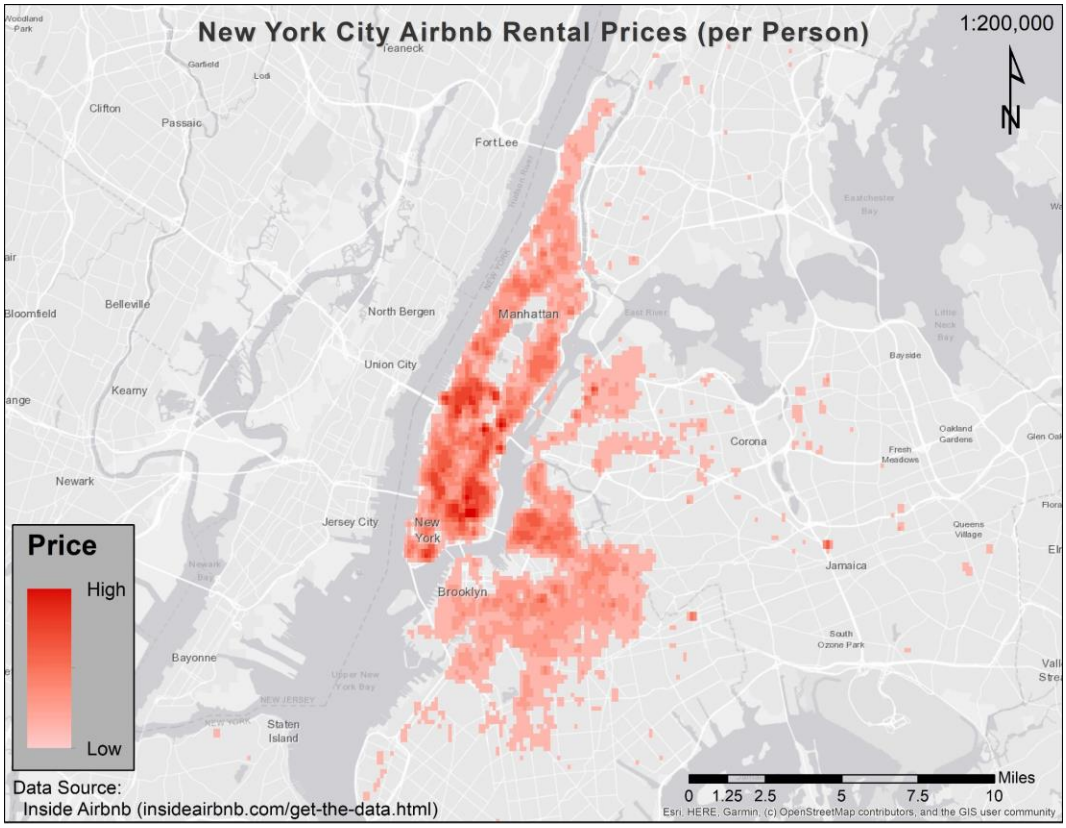
Airbnb Price Prediction

Motivation

Pricing a rental property is a difficult task in a dynamic, ever-fluctuating market. Renters work hard finding the balance between profit and low costs to attract customers. Their business is influenced by their prices which determine the amount of traffic their property receives. Customers, at the same time, have the challenging job of assessing listed prices with little to no knowledge of the local rental property market. From this problem set, the internet company



Airbnb was born. It attempts to streamline the rental process by providing an online rental marketplace. For this market to work, Airbnb needs a way to determine optimal local rent prices, which is where price prediction comes into play. The goal of this project is to develop a price prediction model, using machine learning techniques, to assist renters and customers in cost

evaluation while also providing insight into spatial price distributions. This prediction model is created using two methods: K-Nearest Neighbors and neural networks. It's predictors include spatial location, number of bedrooms/bathrooms, and amenities.

Problem Definition

The developed code attempts to solve several different problems, the first of which is a New York City Borough classification problem. The question here is: in what borough is a spatial location classified? The inputs of this model are the latitude and longitude coordinates which are standardized before use. The output of the model is a number which represents the borough the model believes the coordinate is located in. Constraining the model is the fact that the coordinate locations come from web-scraped Airbnb data. This means our model had to use whatever location the renter decided to pick as their Airbnb listing and could not sample all the locations in a borough and along its boundaries. The borough classification model used a loss of sparse categorical cross entropy with the goal of minimizing the loss.

The second problem involves predicting price values of Airbnb rentals in New York City. Inputs into the model were latitude and longitude coordinates, as well as rental data such as number of bedrooms, number of bathrooms, and number of amenities. The output is a single number prediction of the price. The data is constrained to data in the limits of NYC from the research year of 2020. The goal of the model is to minimize the loss of mean squared error.

Methods

We used two main methods to create prediction models. The first method was a neural network and the second was a K nearest neighbor regressor. For borough classification we used a

neural network with one input layer, five hidden layers and one output layer. Hidden layers used a RELU activation, while the output layer used SoftMax activation. To classify the price values, we used a neural network with one input layer, 10 hidden layers, and one output layer with one node. The hidden layers had a RELU activation function, while the output layer had a linear activation function. We also included dropout layers between the hidden layers.

Neural networks are extremely powerful tools that can extract complex relationships between features. As such we chose this model for our regression analysis because of the complicated relationship amongst location, amenities, and price. The downside of using a neural network is that it can take lots of data and computing power to train. In addition, the results are less understandable for humans.

For each task, price per person prediction and price per person average prediction, we built two neural network models. In the first model we split the data by borough and built a separate neural network for each one. In the second model we fed the entire training dataset into the neural network.

Early on in the development of this project we split the price per person values into several bins, and had the neural network predict which bin the location resided in. The pro of this model was that there were fewer output values to manage which may result in better classification results. However, creating price bins brings up other issues. Because the distribution of prices is not uniform, creating bins that occur with equal probability is difficult. This means the model misclassified many points as it only chose the bin with the most points.

In addition to the neural networks, we also used a K nearest neighbors regressor. In the price per person regression problem, we used a custom distance function which was equal to the distance between locations plus twice the distance of the other rental data variables. For the price

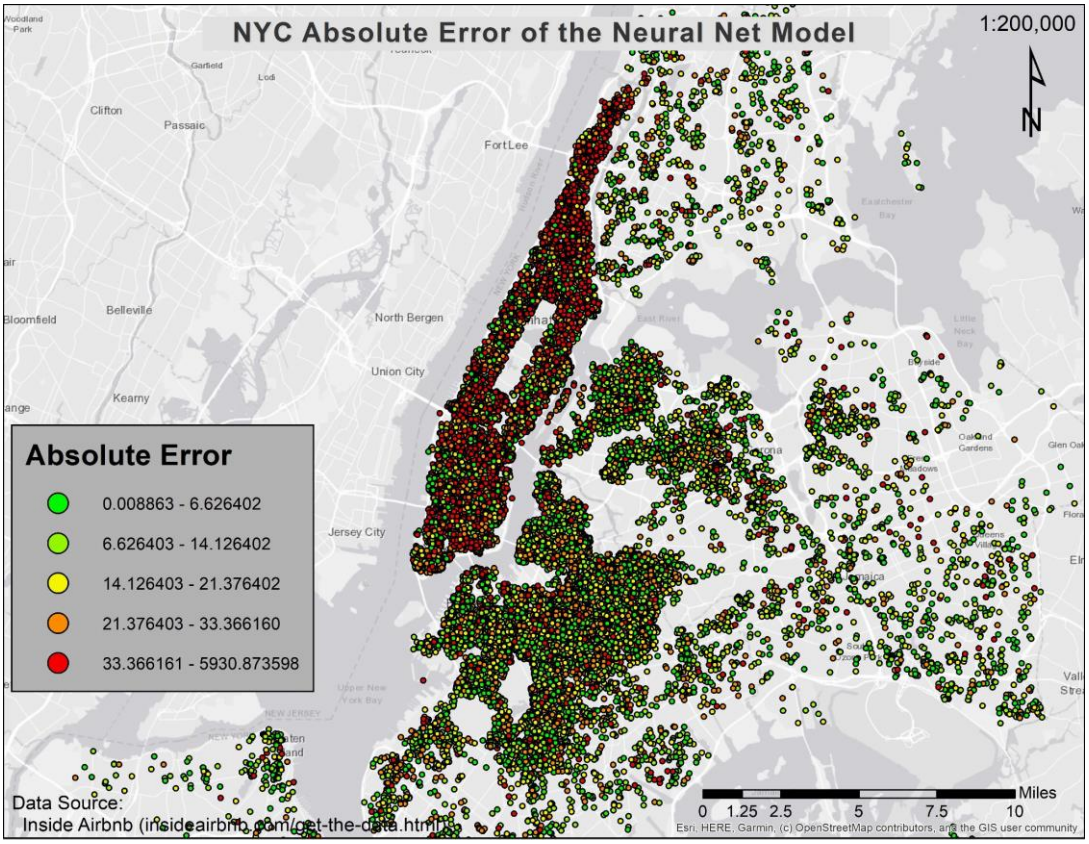
per person average regression problem, we used the default distance function. For the spatial decomposition we used the provided balltree and set K to equal 4 for both models.

K nearest neighbor regressors have the advantage of being very simple and understandable for human observers. They can also be very powerful regression models but have the downside of requiring query data to be close to training data for accurate results.

Data and Results

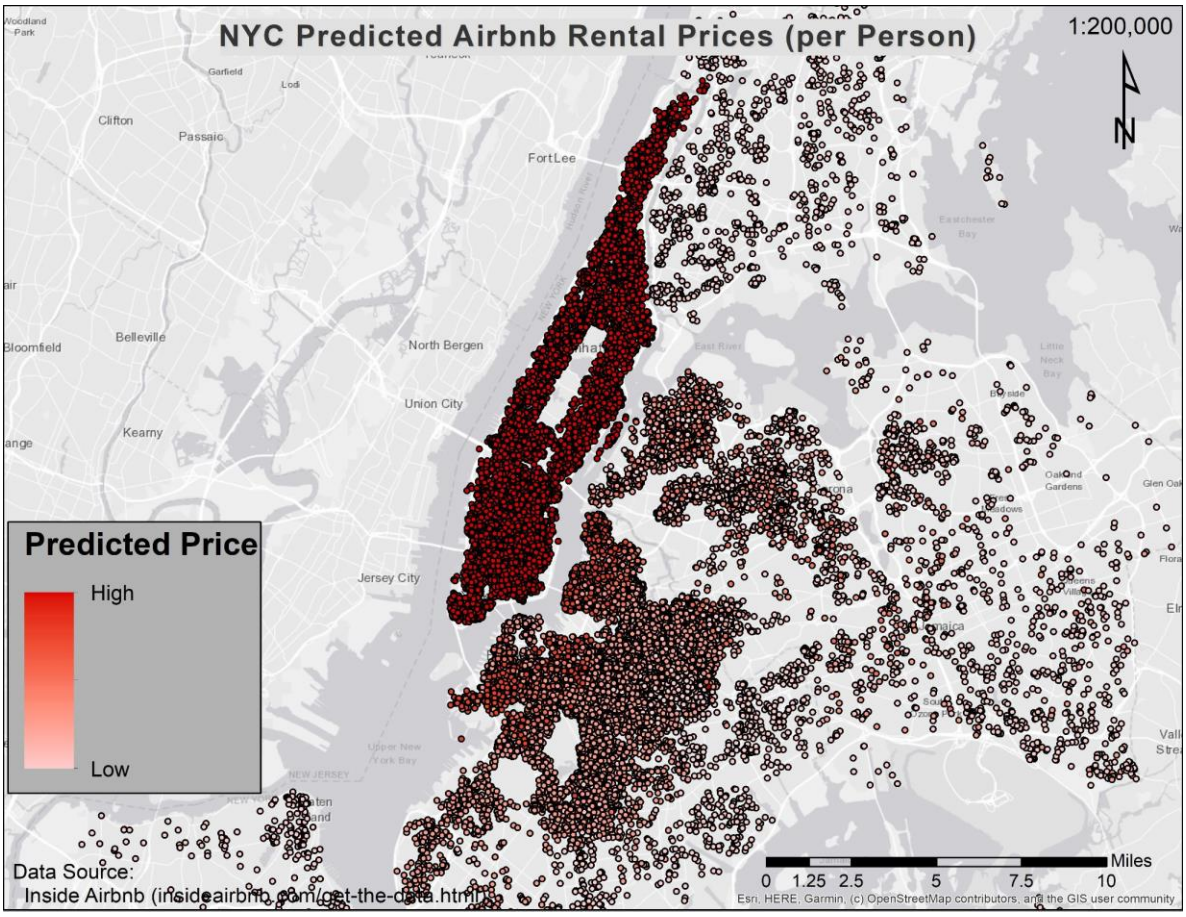
Our data size began as 37012 records with 74 attributes ranging from review scores to descriptions. We decided to cut down the model attributes to just location (lat/long), number of bathrooms, number of bedrooms, borough, and amenities. We assessed these five parameters to be the most influential in determining price and stuck with only five to simply the model.

The first neural network model we ran was the price per person prediction for each borough. The mean absolute percentage error accuracies were Brooklyn = 1.811, Manhattan = 2.465, Queens = 1.452, Staten Island = 1.755, Bronx = 1.743.



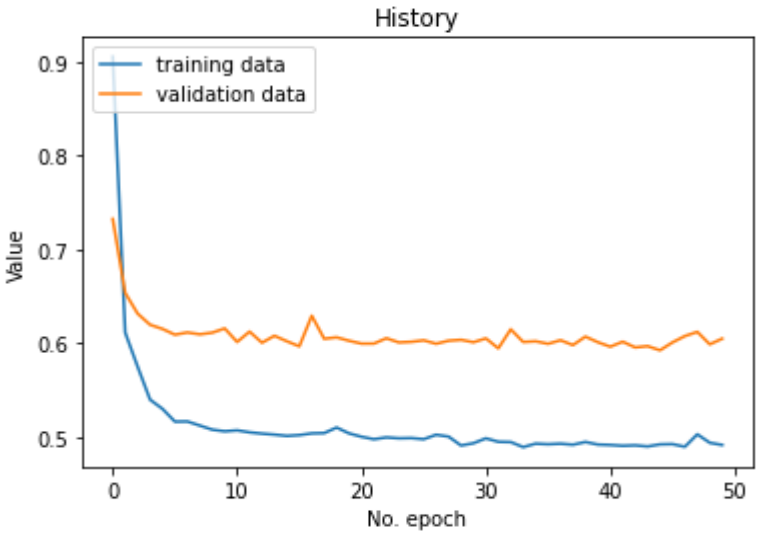
Overall, the model did a respectable job of predicting prices, but it really shined in modeling the spatial distribution of prices. In the ground truth data, the most expensive areas are in downtown Manhattan and North-West Brooklyn which the neural network picked up in its output.

Regarding price prediction troubles, our model had difficulty predicting rental prices specifically

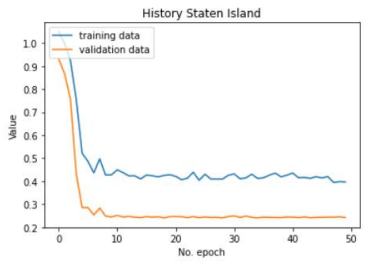


in Manhattan. We assess it struggled in this borough due to its massive variety in rental prices (ranging from $10,000 per person to less than $40 per person), which caused the model to predict the mean price every time.

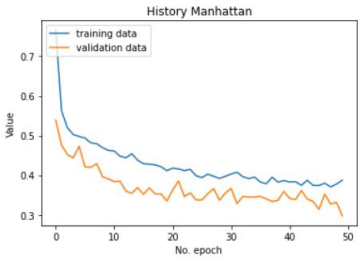
When running the neural network model not broken up by borough, we achieved a mean absolute percentage error of 0.984.



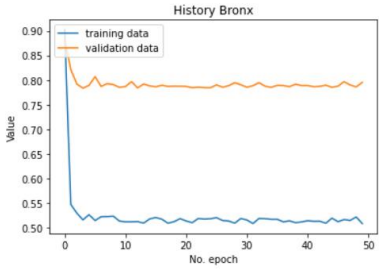
After running the model using the average price per person as the output, we achieved the following mean squared error results (the history graph is the model not broken up by borough):



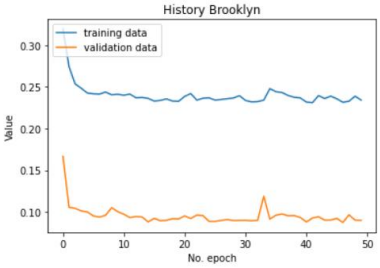
0.2401



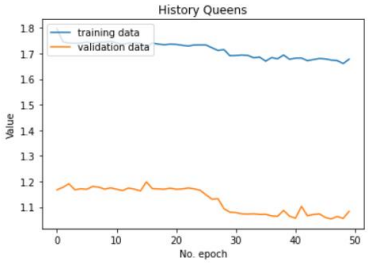
0.2971



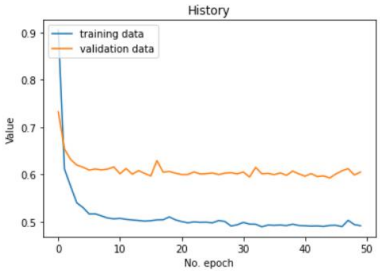
0.7834



0.0880



1.0557



0.5945

Our KNN regressor led to the following results:

- Price per person prediction mean squared error
  - 0.880270720861453
- Average price per person prediction mean squared error
  - 0.1412183334555986

In conclusion, the most accurate price prediction model was our KNN which predicted the average price per person. Throughout this process we realized that the relationship between price and location is more complicated than anticipated. Even in a high end neighborhood in Manhattan with rental prices of $10,000 a day, there could be a vastly cheaper option only a block away. We believe our model may be too simplistic to properly predict rental prices, and that we may have overestimated the weight of spatial location when determining rental prices.