AC209b: Real Estate Price Prediction – Milestone 3

Jiawen Tong, Shiyun Qiu, Yiqi Xie, Chia Chi (Michelle) Ho

April 20, 2018

1 Problem Statement

Real estate purchases is one of the most substantial investment one can make in life, and the real estate market constitutes a significant part of the overall economy. Therefore, the ability to accurately predict real estate prices and trends is lucrative and valuable. For this project, we aim to build a predictive model that accurately forecasts the sold price and the number of days on market of real estate properties in Boston. Many factors affect real estate prices. Thus, the specific goals of our project are as follows:

- Curate historical real estate transaction data with an emphasis on property images
- Develop feature extraction methods and identify key factors that determine property prices
- Build a predictive model to predict sold price and the number of days on market

2 Data Description

2.1 Zillow Data (Provided)

There are 45 columns of fields in the provided .csv file. The key fields are 'MLSNUM', 'STATUS', 'LISTDATE', 'ADDRESS', 'CITY', 'STATE', 'ZIP', 'BEDS', 'BATHS', 'SQFT', 'AGE', 'LOTSIZE', 'SQFT', 'AGE', 'LOTSIZE', 'GARAGE'

2.2 Redfin Data (Externally Curated)

There are 33 columns of fields that we scraped. The key fields are 'beds', 'baths', 'sqft_finished', 'sqft_unfinished', 'year_built', 'year_renovated', 'parking_space', 'garage_space', 'school_ratings', 'school_distances', 'walk_score', 'transit_score', 'bike_score', 'num_photo', photos posted on Redfin.

Note: We did not include string or text fields such as 'REMARKS' for this milestone. We plan to use natural language processing tools such as Latent Dirichlet Allocation (LDA) to process these features in the last milestone.

3 EDA

We focused on exploring variables scraped from Redfin for this exploratory data analysis. Based on the plots below, we found:

- Both response variables, price and DOM, are very right skewed. A single log transformation makes the price distribution much more symmetric while DOM requires 2 sequential log transformation. We will use log(price) and log(log(DOM)) as our response variables.
- For school information scraped from Redfin: A general positive correlation between price and school rating. A general negative correlation between price and school distance.
- For number of photos scraped from Redfin: A general positive correlation between price and number of photos
- For convenience scores scraped from Redfin: A general positive correlation between price and convenience scores

3.1 Exploring Price

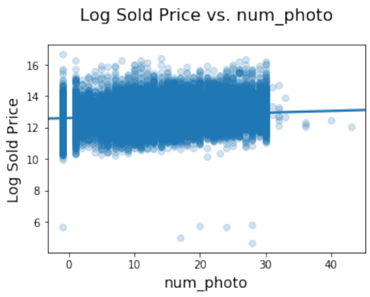

Figure 1: Price VS. Number of Photos

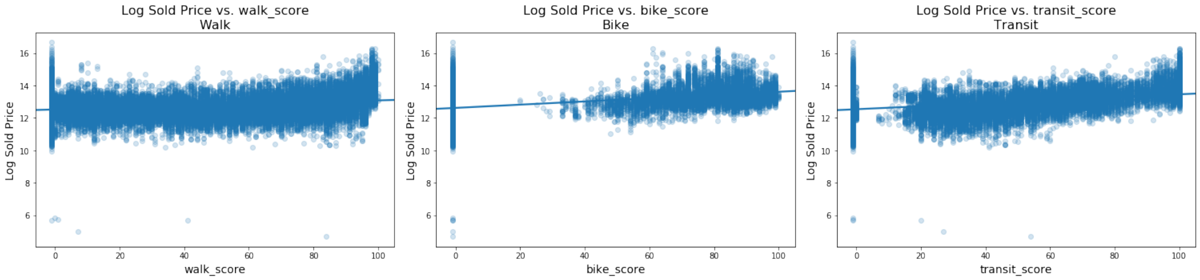

Figure 2: Price VS. Convenience Scores

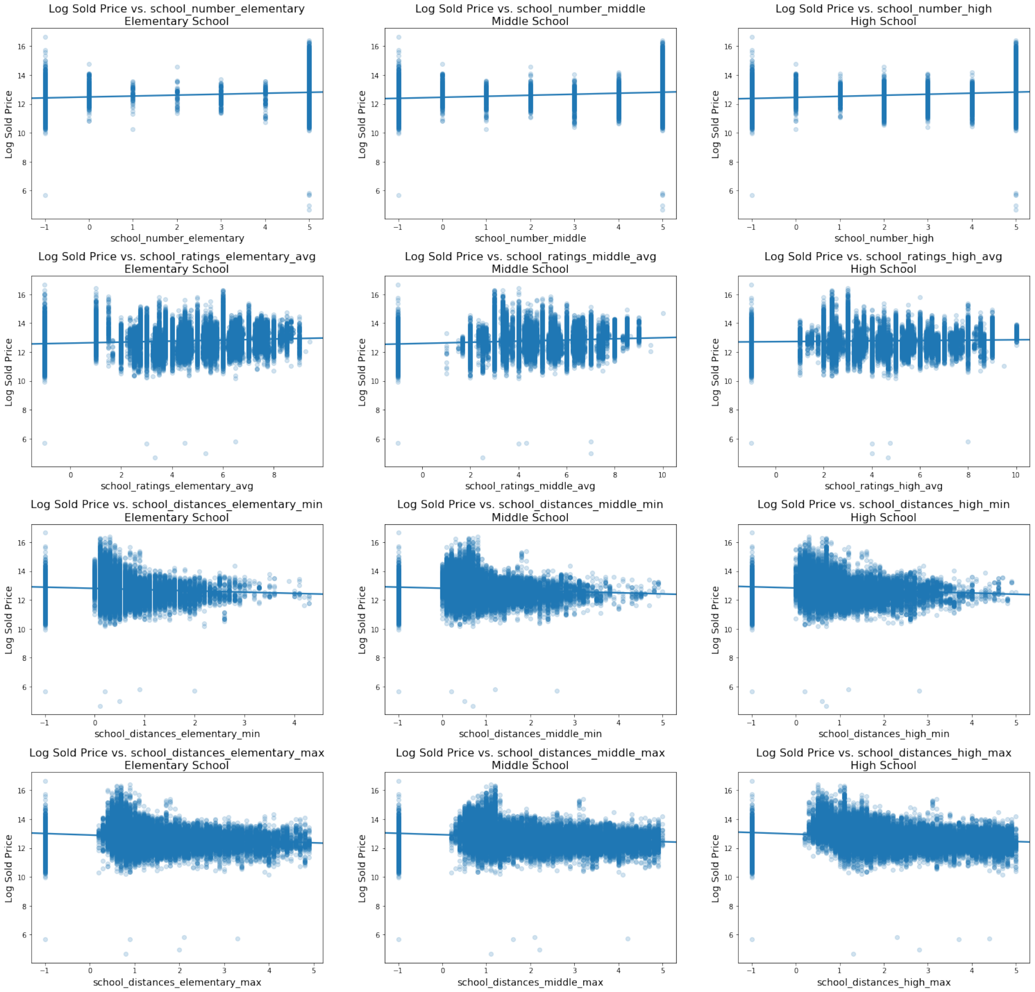

Figure 3: Price VS. School

3.2 Exploring DOM

Similar but less obvious trends are observed with DOM. See more EDA graphs on our github.

4 Approach

4.1 Data Preprocessing/Feature Extraction

1. Convert (city, state, zip code) combinations into (longitude, latitude) via the Google Map Geoencoding API – This produces 2 features.
2. Extract features from images
    - Utilize keras.ResNet50 and pretrained imagenet weights as our model.

   - For each image, extract a 2048-dimensional feature vector by obtaining the output of the last layer (`flatten_1`) prior to the final prediction layer in the model.
   - For each house, take the average features of all of its images as the final image features used to predict sold price.

   This produces 2048 features.

3. Extract features from top 5 closest school ratings/distances
   - Record average of school ratings and distances
   - Record number of closest schools
   - Record the minimum and maximum distances from the house to the schools

   This produces 5 features.

4. Merge Zillow/Redfin data and drop listings that are not in Massachusetts
5. Take the Redfin features if they overlap with the zillow ones (e.g., BEDS, BATHS, SQFT)
6. Replace all unreasonable values (e.g. AGE = -8000) and/or missing data with -1
7. Perform PCA on the standardized image features with 40 components (explained over 90% of variance)

4.2 Feature Sets

- Zillow (10): The provided Zillow data
- Zillow (10) + Redfin (20): Merged Zillow data and curated Redfin data
- Zillow (10) + Redfin (20) + Img-PCA (40): Zillow and redfin data plus the first 40 principal components of the 2048 image features.

4.3 Models

- Ridge Regression
- XGBoost (eXtreme Gradient Boosting)
- LightGBM

5 Preliminary Results

For this milestone, we ran the models on Condo properties. Summarized as below, all three models yield better performance with additional features scraped from Redfin (i.e school ratings, walk score and transit scores). We also observed that the additional image features significantly improved the Ridge model but not much improved XGBoost and LightGBM. We suspect that a better way to use the image features would be to build a separate predictive model with image features only, which is then stacked with other models fitted on the rest of the features.

Training R^2			
	Zillow	Zillow + Redfin	Zillow + Redfin + Img-PCA
Ridge	0.3287	0.6383	0.7401
XGBoost	0.9758	0.9866	0.9785
LightGBM	0.9571	0.9833	0.9755

Test R^2			
	Zillow	Zillow + Redfin	Zillow + Redfin + Img-PCA
Ridge	0.3106	0.6292	0.7347
XGBoost	0.9390	0.9453	0.9359
LightGBM	0.9381	0.9476	0.9423

6 Future Plans

- Scrape information and download images for all properties
- Develop a better method to incorporate image information
- Explore the text features in the given dataset
- Apply boosting models to all three property types
- Use Random Forests, Neural Networks, stacking and other models
- Predict price and days on market for all properties in the test set