# House Price Prediction in Atlanta

## Introduction

In the U.S. real estate market, house price is a crucial factor that influences both economical contributions and people's living standards. Therefore, having a model that can predict house prices is beneficial. Although models for house price prediction have been researched and evaluated extensively in the past, an interactive web based system that allows users to navigate the predicted prices based on geological location, house types, and other information is not present. Our goal for this project is to build an interactive web application that predicts the house prices in the Atlanta Metropolitan Area, using and comparing different machine learning models trained using data we have collected from various sources. We will then evaluate both performances of the model and the user experience while interacting with the system.

Based on the previous works in related fields, our approach involves the following innovations:

1. interactive map
2. comparison between different ML models
3. innovative visualization
4. more features for prediction

## 1 Survey

The first feature we will include in our model is location and commuting cost. One classical model is standard urban model (SUM), which assumes one center for a city and correlate the house price with the distance to the center of the city [1]. Larger cities have higher house price volatility due to lower elasticity of resources and there is higher spatial heterogeneity for the submarkets in larger cities [2]. In suburban regions, the house supply has high elasticity and the price tends to be stationary [3]. The house price prediction model can be implemented in a location-based recommendation system (RS). Such system provides predicted values based on location information and hence increases accuracy [4]. Park et al. [5] developed a RS taking in spatial variation using Bayesian Network [6, 7] which can efficiently provide the conditional probability distributions of results. Further detail about implementation will be presented in Part 2.

## 2 Methodology

### 2.1 Data Collection and Preprocessing

To efficiently gather real-estate data, a web scraper is built using Python to extract the data from zillow.com. The scraper extracts details of property listings based on zip code, then imports the information into a csv file. The property details include property location (e.g. City, State, longitude and latitude etc.), features (i.e., number of bedrooms and bathrooms and sqft), price and related URL links. After scraping all the listings on the first page, a new URL of next results page will be generated based on the information stored in the web elements. Data on the new page will be extracted. These scraping procedures will continue for all zip code.

Collected data is cleaned and extracted through OpenRefine. Useless columns are removed. Further information of postcode and corresponding latitude / longitude is incorporated to the original data by joining multiple tables.

## 2.2 Front-end Development

We are using HTML, CSS, and JavaScript to develop the user interface for this web application. So far, Bootstrap framework has been employed for easier and faster front-end development. Another library used to develop the front-end is Material Design for Bootstrap (MDB), which is mainly based on Bootstrap and provides a collection of flexible components. At this stage, we try to keep the interface simple but functional and efficient. More graphical features will be added.

The user interface mainly consists of two parts: the input section and the output section. The input section includes an input panel on the right to take input values from users, and an interactive map on the left, which enables users to pick a desirable point to search. The Google Maps JavaScript API has been employed to create and tailor such interactive map for the purpose of this project. By clicking on any point on this map, a marker surrounded by a modifiable circle will appear, and the correspondence information (latitude and longitude of the point, and the specified search radius) will be prepared to send to the back-end. By pressing the "Search" button, all the information gathered in the input section will be transferred to the backend for processing. As soon as the process is completed, the results will be posted on the output section.

We will also implement a visualization using d3. After accomplishing the predictive model, a set of illustrative d3 data visualizations will be augmented to this section to demonstrate the results in a more descriptive way.

## 2.3 Back-end Establishment

Backend is developed using Flask, which gets data from front-end in two ways. User can get estimated price by either inputting zip code or pinning any point on the map.

After receiving data, back-end calls machine learning program. In this program, multiple machine learning algorithms are employed and compared. The program then returns jsonified results, including prediction price and model parameters to back-end program, and then to the "results" column on the Web page.

Backend also incorporates wtform module to restrict input. For illegitimate input, the website will not proceed but instead flash specified error message.

## 2.4 Machine Learning

As for the price prediction model, we start with toy data set for machine learning training. The data includes hundreds of regions in Great Atlanta region. Currently we only consider two features, longitude and latitude, and the prediction target is the selling price.

The first machine learning model we will implement is linear models. Two common methods for calibrating linear regression are least squares estimation (LSE) or maximum likelihood estimation

(MLE) [8]. House price depends on both location and time. Spatial effects include two parts, spatial dependence and spatial heterogeneity. For spatial effects, locations are grouped into sub-regions [9]. STAR model [10], which is based on Hedonic regression and considers the spatial and temporal factors, could perform better than ordinary least square model. House prices might vary even under the same conditions. Thus fuzzy linear regression is introduced to handle the fuzziness of such systems[11].

Then, several other models, including Random Forest, Ridge Regression, Neural Network, and Gradient Boosting have been trained and compared. Vladimir et al. [12] proved that Random Forest is powerful generating accurate results when making prediction. However, since it's hard to see the exact structure of each tree, we cannot intuitively see the relationships between features and predictions [13]. As for Ridge Regression, [14] it can proficiently avoid overfitting by adding penalty, also Marquardt et al. [15] displayed the "ridge trace" to better tune parameters based on sensitivities between coefficients and data sets. But these articles also show the shortcoming that there still exist impacts of irrelevant features since it doesn't enforce $\boldsymbol{\beta}$ coefficient to 0. Neural network is robust to error and gives good performance as a prediction model according to works by Odom [16], while the tradeoff is larger computational burden and higher possibility to overfitting [17]. Gradient Boosting keeps searching for steepest descent to minimize loss function, which is an efficient way [18]. However, this article mostly concentrate on mathematical theory instead of practical using. Thus, we will apply these models to this prediction task to see how they work.

Two methods were used to estimate the quality of trained model, one is to split the data set to train and test set, another is by cross validation. The test data is 10% of the total data set.

Right now we have explored only using Latitude and Longitude as the training features. After we have confirmed on the feature set, we will perform a full comparison among those models and decide which model to use in the final version.

# 3 Experiments and Evaluation

Up until this point, we have experimented with 4 ML models. We have used cross validation to split the dataset into 90% training and 10% testing sets. The results are shown in Table 3.1.

Table 3.1 Performances of 4 ML models

| | Random Forest | Ridge Regression | Neural Network | Gradient Boosting |
|---|---|---|---|---|
| Parameters | #estimators: 20 | alpha: 0.1 | #hidden layers: 20 | #estimators: 20 |
| R2 score | 0.575 | 0.171 | -8.631 | 0.550 |
| train score | 0.910 | 0.124 | -7.982 | 0.906 |
| testing score | 0.108 | -0.007 | -8.255 | 0.122 |

In the future, as more data are collected, we will perform our ML models using more relevant features. The accuracies of the models will be obtained using cross validation and the predictions of the most suitable ones will be aggregated and applied to our application. We will also use the predicted price index obtained from our model to compare with the data from the FHFA dataset as the groundtruth.

To evaluate user experience, we will ask people outside of our team to interact with the program, while monitoring their navigation and experience when using the system. Some survey questions regarding the interface experience will also be asked. We will also test the reliability of the system to make sure our implementation is bug-free.

# 4 Working Plan and Distribution

Plan of activities are shown in table 4.1. All team members contributed similar effort.

Table 4.1 Working Plan and Distribution

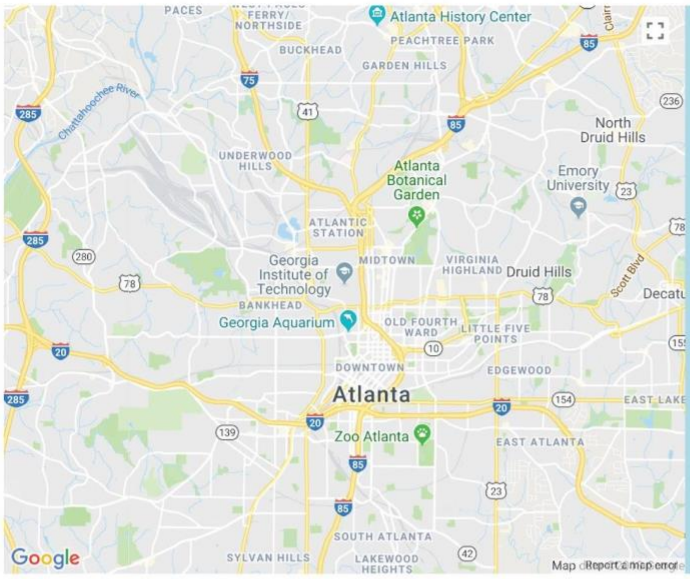
| Group Member | Old Responsibilities | Revised Responsibilities |
|---|---|---|
| Zichen Wang | Machine Learning | ML, Report Organization |
| Wenqing Shen | ML Implementations | ML Implementations |
| Yixing Li | Back-end | Data Cleaning, Back-end, Code Coordination |
| Dong Gao | Data Cleaning, Scraping | Data Cleaning, Scraping |
| Xiangyi Yan | Data Visualization | Data Visualization, Report Organization |
| Shahrokh Shahi | Front-end | Front-end |

# 5 Screenshots

Two screenshots taken from the workable web-page are shown below, one shows the initial website and the other shows when website of a successful price inquiry.

# ATLANTA HOUSE PRICE PREDICTION

The User Interface Demo for House Price Prediction Model Group Project




zipcode

num3ed

num3ath

Search my next home!

Results | Visualization

Outputs:

DETAILS

## House Price Prediction

House price prediction is a platform implementing various machine learning mehods to predict house price in Atlanta

**DATA AND VISUAL ANALYTICS | GROUP PROJECT**

House price prediction is a platform implementing various machine learning mehods to predict house price in Atlanta

**HOUSE PRICE PREDICTION**

MDBootstrap

MDWordPress

BrandFlow

Bootstrap Angular

**CONTACT**

Georgia Institute of Technology, GA 30332, US

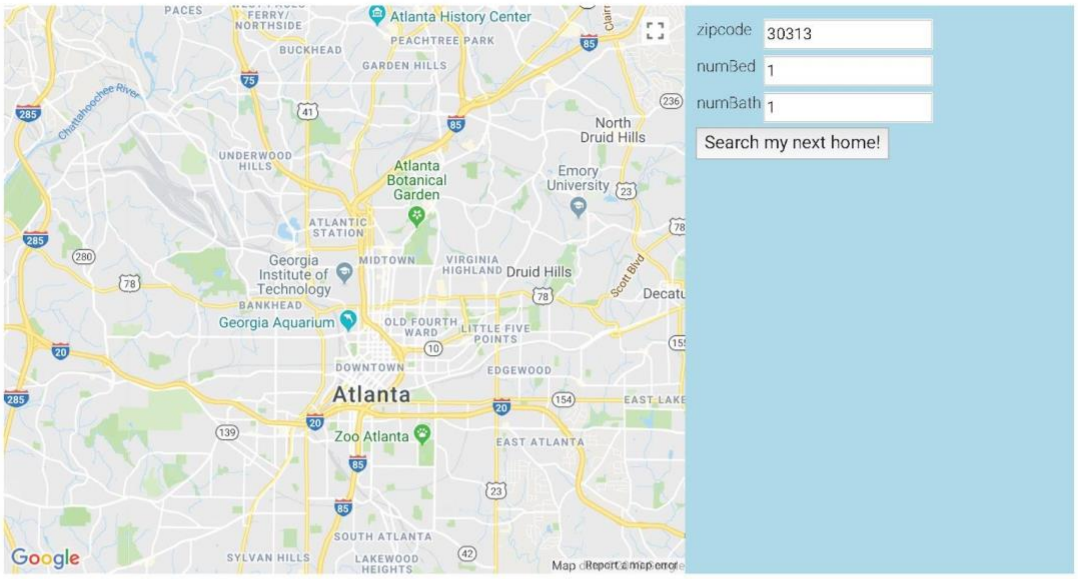
xxxx@gatech.edu

+ 1 404 xxx xxxx

+ 1 404 xxx xxxx

# ATLANTA HOUSE PRICE PREDICTION

The User Interface Demo for House Price Prediction Model Group Project




Results Visualization

Outputs:

[329950.0]

DETAILS

## House Price Prediction

House price prediction is a platform implementing various machine learning mehods to predict house price in Atlanta

**DATA AND VISUAL ANALYTICS | GROUP PROJECT**

House price prediction is a platform implementing various machine learning mehods to predict house price in Atlanta

**HOUSE PRICE PREDICTION**

MDBootstrap

MDWordPress

BrandFlow

Bootstrap Angular

**CONTACT**

Georgia Institute of Technology, GA 30332, US

xxxx@gatech.edu

+ 1 404 xxx xxxx

+ 1 404 xxx xxxx

Reference:


1. Muth R. *Cities and housing: The spatial patterns of urban residential land use.* University of Chicago, Chicago. 1969;4:114-23.
2. Bogin, A., W. Doerner, and W. Larson, *Local house price dynamics: New indices and stylized facts.* Real Estate Economics, 2016.
3. Glaeser, E.L., et al., *Housing dynamics: An urban approach.* Journal of Urban Economics, 2014. **81**: p. 45-56.
4. Horozov, T., N. Narasimhan, and V. Vasudevan, *Location-based recommendation system*. 2006, Google Patents.
5. Park, M.-H., J.-H. Hong, and S.-B. Cho. *Location-based recommendation system using bayesian user's preference model in mobile devices*. in *International Conference on Ubiquitous Intelligence and Computing*. 2007. Springer.
6. *Bayesian network.* February 28, 2018]; Available from: https://en.wikipedia.org/wiki/Bayesian_network.
7. Friedman, N., D. Geiger, and M. Goldszmidt, *Bayesian network classifiers.* Machine learning, 1997. **29**(2-3): p. 131-163.
8. Seber, G.A. and A.J. Lee, *Linear regression analysis*. Vol. 329. 2012: John Wiley & Sons.
9. Liu, X., *Spatial and temporal dependence in house price prediction.* The Journal of Real Estate Finance and Economics, 2013. **47**(2): p. 341-369.
10. Pace, R.K., et al., *Spatiotemporal autoregressive models of neighborhood effects.* The Journal of Real Estate Finance and Economics, 1998. **17**(1): p. 15-33.
11. Asai, H.T.-S.U.-K. and S. Tanaka, *Linear regression analysis with fuzzy model.* IEEE Transaction Systems Man and Cybermatics, 1982. **12**(6): p. 903-07.
12. Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *Journal of Chemical Information and Computer Sciences,43*(6), 1947-1958. doi:10.1021/ci034160g
13. Wiesmeier, M., Barthold, F., Blank, B. et al. Plant Soil (2011) 340: 7. https://doi.org/10.1007/s11104-010-0425-z
14. Hoerl, A. E., & Kennard, R. W. (1970). Ridge Regression: Applications to Nonorthogonal Problems. *Technometrics,12*(1), 69-82. doi:10.1080/00401706.1970.10488635
15. Marquardt, D., & Snee, R. (1975). Ridge Regression in Practice. The American Statistician, 29(1), 3-20. doi:10.2307/2683673
16. Odom, M., & Sharda, R. (1990). A neural network model for bankruptcy prediction. *1990 IJCNN International Joint Conference on Neural Networks*. doi:10.1109/ijcnn.1990.137710
17. Tu, J. V. (1996). Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes. *Journal of Clinical Epidemiology,49*(11), 1225-1231. doi:10.1016/s0895-4356(96)00002-9
18. Death, G. (2007). Boosted Trees For Ecological Modeling And Prediction. *Ecology,88*(1), 243-251. doi:10.1890/0012-9658(2007)88[243:btfema]2.0.co;2