Prediction on Housing Price Based on Deep Learning

Chaoyue Zheng(cz2529) and Chen Wang(cw3119)

Columbia University

1 Introduction

Housing price is of great significance to a modern city, reflecting the booming and decay of the economy. An accurate prediction can be great useful for the homeowners, buyers, agencies, developers and investors. However, traditional predicting process mostly uses Multi Linear Regression (may be time series method) or Stochastic Process prediction, which turns out to be less valuable accuracy. To address this issue, we applied non-linear algorithms like DNN, Random Forest and XGBoost to better fit the dataset by analyzing the feature attributes of the price. We also select Linear Regression as the benchmark to evaluate these models by MSE. This paper focuses the house sales market in Manhattan. Our contributions mainly include (1) Scraping information about rental/sales housing price, transportation information and restaurants from real estate websites, NYC.gov, Google Map, and using longitude and latitude to build a comprehensive database in the NYC areas. (2) Predicting Manhattan house sales prices using Linear Regression, DNN, XGBoost, Random Forest, which XGboost beats others with lowest MSE.

2 Prediction Model

2.1 Scraping and Analysis of housing price influencing factors

2.1.1 Access to Real Estate Data with Web Crawler Technology

Large volume and high-quality data is required to train these algorithms, which is the fundamental part of building a reliable model. Our data source includes Real Estate websites (Douglas Elliman, Compass, New York Times Real Estate section) and Google Map. It is hard to simply download from these websites, so we scraped using web crawler. We scraped house information for New York City (house price, number of bathrooms and bedrooms, house website, property type, latitude and longitude) in Real Estate websites and neighborhood information (restaurant level, subway information) with Google Map API. For the restaurant level, we average the price level of the top 20 restaurants around each house. We also use the following formula to compute the distance between house and nearest subway station to measure the transportation convenience.

2.1.2 Analysis of Housing Price Influencing Factors

We choose 6 influencing factors, bathroom, bedroom, size, subway distance, restaurant level and property type. As can be seen in the table, the 75 percentiles for bathroom, bathroom and size is relatively reasonable, while the max value like 39, 25 may be somehow confusing. In fact, from further research, these data come from a whole building for sale, not a single apartment. Restaurant level ranges from 1 to 4, and the data shows it roughly forms a normal distribution. We also obtained data for property type, as shown in the histogram, multi-family type is the majority, three times higher than the second type Condo.

2.2 Data Processing

We combined all the houses from three Real Estate website together and formed a 11,583 sample. In this sample, we detected some fields are missing. Since this data set is relatively large enough and different method for filling in the missing values may result in large difference in final model prediction. We simply delete the rows with missing values. Besides, property type is a categorical feature, so we encode this feature using one-hot encoding. Finally, we get a sample with 9482 rows and 19 columns for training.

Table 1. Data Description

	bathroom	bedroom	size	subway distance
mean	2.50	2.40	15333.09	0.02444
std	1.56	1.57	18038.17	0.29765
min	0.00	0.00	210.00	0.00000
25%	2.00	2.00	4352.50	0.00056
50%	2.00	2.00	10295.00	0.00310
75%	3.00	3.00	18750.00	0.01319
max	39.00	25.00	248470.00	9.66390

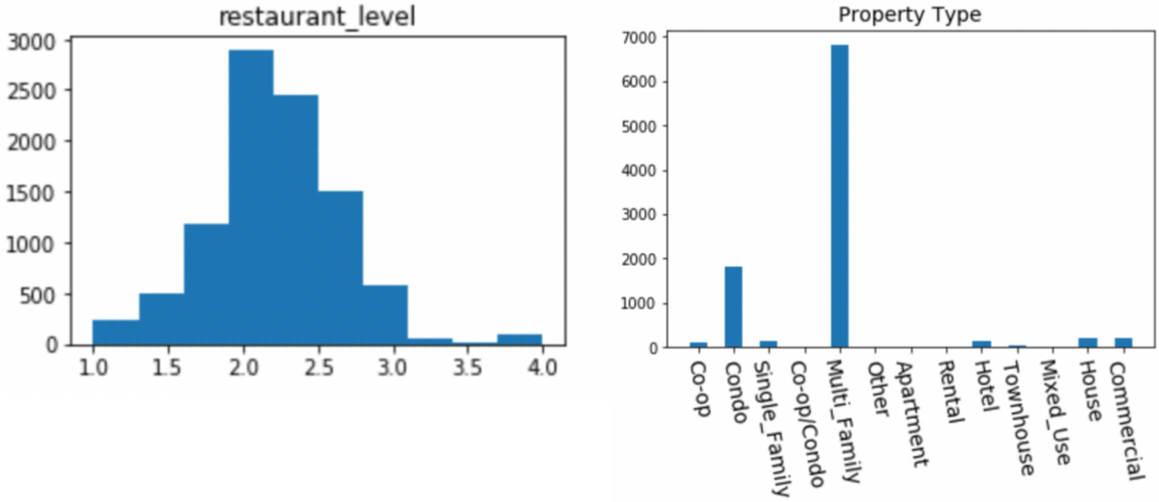
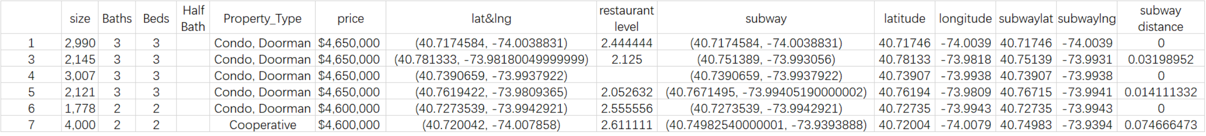
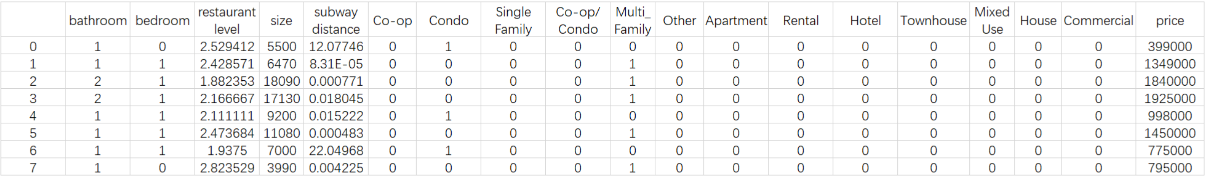


Fig. 1. Data Visualization for Restaurant Level and Property Type



	size	Baths	Beds	Half Bath	Property_Type	price	lat&lng	restaurant level	subway	latitude	longitude	subwaylat	subwaylng	subway distance
1	2,990	3	3		Condo, Doorman	$4,650,000	(40.7174584, -74.0038831)	2.444444	(40.7174584, -74.0038831)	40.71746	-74.0039	40.71746	-74.0039	0
3	2,145	3	3		Condo, Doorman	$4,650,000	(40.781333, -73.98180049999999)	2.125	(40.751389, -73.993056)	40.78133	-73.9818	40.75139	-73.9931	0.03198952
4	3,007	3	3		Condo, Doorman	$4,650,000	(40.7390659, -73.9937922)		(40.7390659, -73.9937922)	40.73907	-73.9938	40.73907	-73.9938	0
5	2,121	3	3		Condo, Doorman	$4,650,000	(40.7619422, -73.9809365)	2.052632	(40.7671495, -73.99405190000002)	40.76194	-73.9809	40.76715	-73.9941	0.014111332
6	1,778	2	2		Condo, Doorman	$4,600,000	(40.7273539, -73.9942921)	2.555556	(40.7273539, -73.9942921)	40.72735	-73.9943	40.72735	-73.9943	0
7	4,000	2	2		Cooperative	$4,600,000	(40.720042, -74.007858)	2.611111	(40.74982540000001, -73.9393888)	40.72004	-74.0079	40.74983	-73.9394	0.074666473

Fig. 2. Sample Original Data



	bathroom	bedroom	restaurant level	size	subway distance	Co-op	Condo	Single Family	Co-op/ Condo	Multi_ Family	Other	Apartment	Rental	Hotel	Townhouse	Mixed Use	House	Commercial	price
0	1	0	2.529412	5500	12.07746	0	1	0	0	0	0	0	0	0	0	0	0	0	399000
1	1	1	2.428571	6470	8.31E-05	0	0	0	0	1	0	0	0	0	0	0	0	0	1349000
2	2	1	1.882353	18090	0.000771	0	0	0	0	1	0	0	0	0	0	0	0	0	1840000
3	2	1	2.166667	17130	0.018045	0	0	0	0	1	0	0	0	0	0	0	0	0	1925000
4	1	1	2.111111	9200	0.015222	0	1	0	0	0	0	0	0	0	0	0	0	0	998000
5	1	1	2.473684	11080	0.000483	0	0	0	0	1	0	0	0	0	0	0	0	0	1450000
6	1	1	1.9375	7000	22.04968	0	1	0	0	0	0	0	0	0	0	0	0	0	775000
7	1	0	2.823529	3990	0.004225	0	0	0	0	1	0	0	0	0	0	0	0	0	795000

Fig. 3. Sample Cleaned Data

2.3 Model Description

Based on data we cleaned, we have two kinds of features, where the first kind is house-level part, including house size, the number of bedrooms and bathrooms, property type, etc and the second part describes the neighborhood part, such as average restaurants price level around houses and the distance between nearest subway and houses measured by latitude and longitude. In this part, we would like to use four linear or nonlinear models.

2.3.1 Multi-Linear Regression

Firstly, we simply use linear regression model to test the linear relationship between explanatory features and dependent variable.

$$price_i = \beta_0 + \beta_1 * bathroom_i + \beta_2 * bedroom_i + \beta_3 * pricelevel_i + \beta_4 * size_i + \beta_5 * subwaydistance_i + \beta_6 * propertytype_i$$

Table 2 shows the results for multi-linear regression.

Table 2. Results For Multi-Linear Regression

Features	Coefficient	Variance
bathroom	0.433***	0.0180
bedroom	0.120***	0.0323
restaurant level	0.0486***	0.00302
size	0.559***	0.00848
subway distance	-0.0131***	0.00433

2.3.2 Dense Neural Network

Sometimes, linear model cannot well explain the dependent variable. Dense Neural Network, a nonlinear model, is a form of artificial intelligence that consists of a number of interconnected processing neurons, that mimic the functions of biological neurons to process information in parallel. You can choose one or more hidden layers and the number of neurons in each layer. Backpropagation training algorithm is commonly used to train DNNs, which calculate the errors between the predicted output and the target output(actual house price) and back-propagate the error to adjust the connection weights between the neurons in adjacent layers with the aim to find the optimal connections between the neurons that best map the relationships between inputs (e.g. various attributes of individual houses and neighborhood features) and output (e.g. house prices). Fig. 2 shows a sample of neurons in model with two hidden layers.

Although there are lots of studies compared DNN with multiple regression with OLS estimation, there are no consensus whether DNN is better than other regression models. Therefore, the result in this study cannot well verify that DNN certainly beats other models. In this model, attributes of individual houses, such as size, the number of bedrooms etc, are neurons in input layers, while house price is the neuron in output layer. After hyperparameter tuning, we found that the model with 4 dense layers and 128, 256, 256, 256 neurons respectively beats other hyperparameters.

2.3.3 Random Forest

Random forest, an ensemble algorithm employing multiple decision trees, is an algorithm whose basic unit is a decision tree. It is are an improvement over bagged decision trees, showing its superiority in many application areas. Since it is an ensemble algorithm, you can get an unbiased estimation of the internal generation error during the forest generating process, and the generalization capability is good. Nevertheless, random forest may suffer overfitting in some regression problems where noise occurs often.

2.3.4 XGBoost

XGBoost(XGBT), an open-source software library, is a decision-tree-based ensemble Machine Learning algorithm that uses a gradient boosting framework. Compared with other models, we can train models more efficiently and get better prediction results. A benefit using gradient boosting is that after the boosted tree are built, it is relatively straightforward to retrieve importance for each feature.

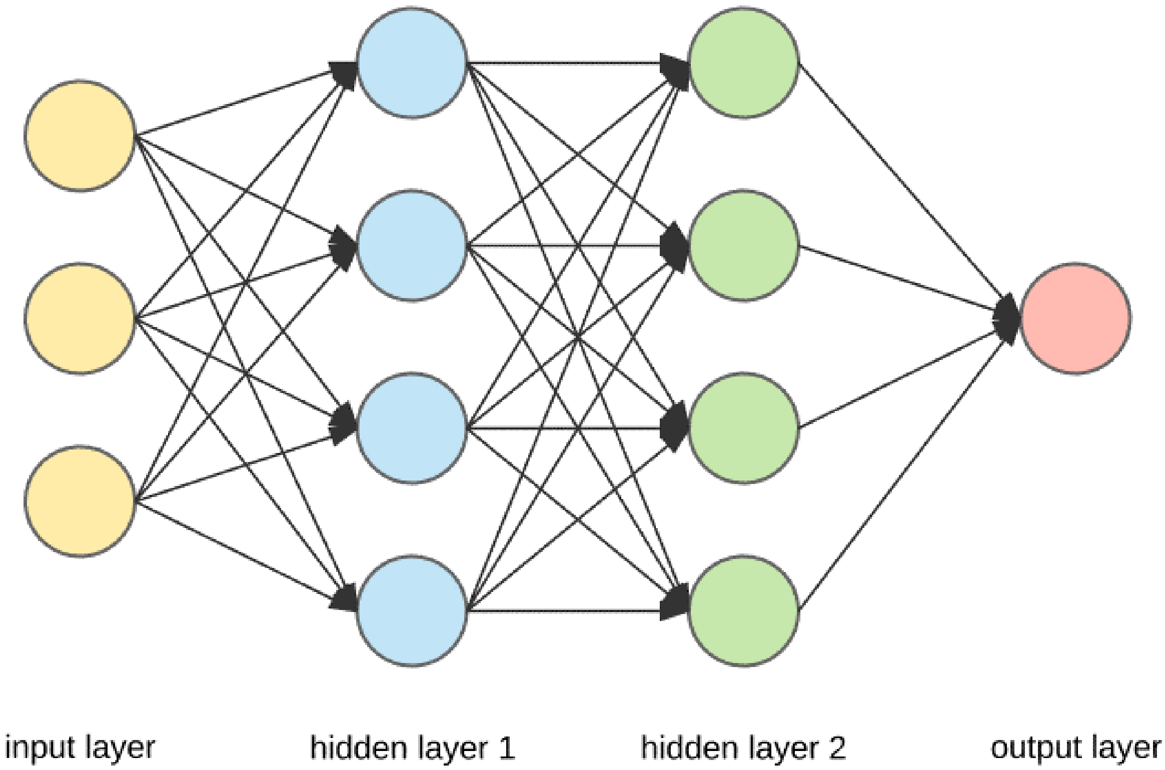


Fig. 4. Neural Network with Two Hidden Layers

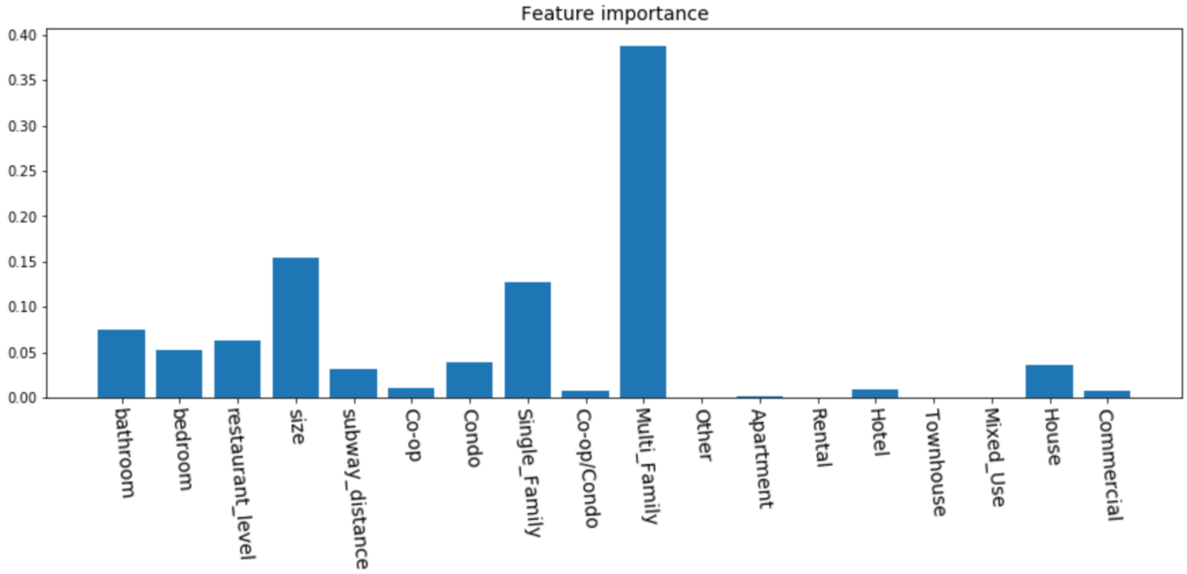


Fig. 5. Feature Importance

2.3.5 Model Parameters

In this case, we need first to train these selected base predictors. In order to select best hyperparameter, the training data is divided into two parts, whereas 80% of it is for training and 20% of it is used for valuation. The parameters which get the best results are used for test data. The parameters to be set for the prediction models are listed as follows:

Table 3. Model Parameters

Prediction Models	Model Params After Training
DNN	n$_{l}ayers : 4$ n$_{n}eurons : 128, 256, 256, 256$ activation function: RELU
Random Forest	n$_{e}stimators : 90$ max$_{d}epth : 13$ min$_{s}amples_{s}plit : 2$ min$_{s}amples_{l}eaf : 2$ random$_{s}tate : 10$
XGBoost	n$_{e}stimators : 1200$ max$_{d}epth : 3$ gamma: 0 colsample$_{b}ytree : 0.9$ subsample: 0.9 reg$_{a}lpha : 0$ reg$_{l}ambda : 0.1$ learning$_{r}ate : 0.1$

2.4 Results

After tuning the hyperparameters, the selected parameters can be used for conducting predictions. The following table lists mean squared error of test data, which can be applied to evaluate the performance of each model. Considering normalizing all features and dependent variables, MSE value is also measured in normalized scale. Comparing loss functions/MSE, we can see that XGBoost beats all other models when predicting house price while linear regression performs worst among all models as predicted. It is not difficult to recognize the prediction results obtained by DNN perform not so well compared with tree models.

Table 4. Mean Square Error

Model	Mean Squared Error
Linear Regression	0.00170884
DNN	0.00125612
Random Forest	0.00093139
XGBoost	0.00068758

Table 4 depicts predicted house prices against actual ones. We select first 100 samples to visualize results. The horizontal line represented the data samples, ranging from 0 to 100, and the vertical axis shows house prices. The Blue one is for actual price, while the orange one is predicted price. Although it is hard to validate the accuracy and effectiveness of each model, we can roughly find the difference between actual price and predicted price are tinier in XGBoost model.

3 Conclusion

Nowadays, house price are influenced by a variety of factors, such as building location, surrounding facilities and characteristics of house itself. In this paper, firstly we mainly used a web crawler to crawl

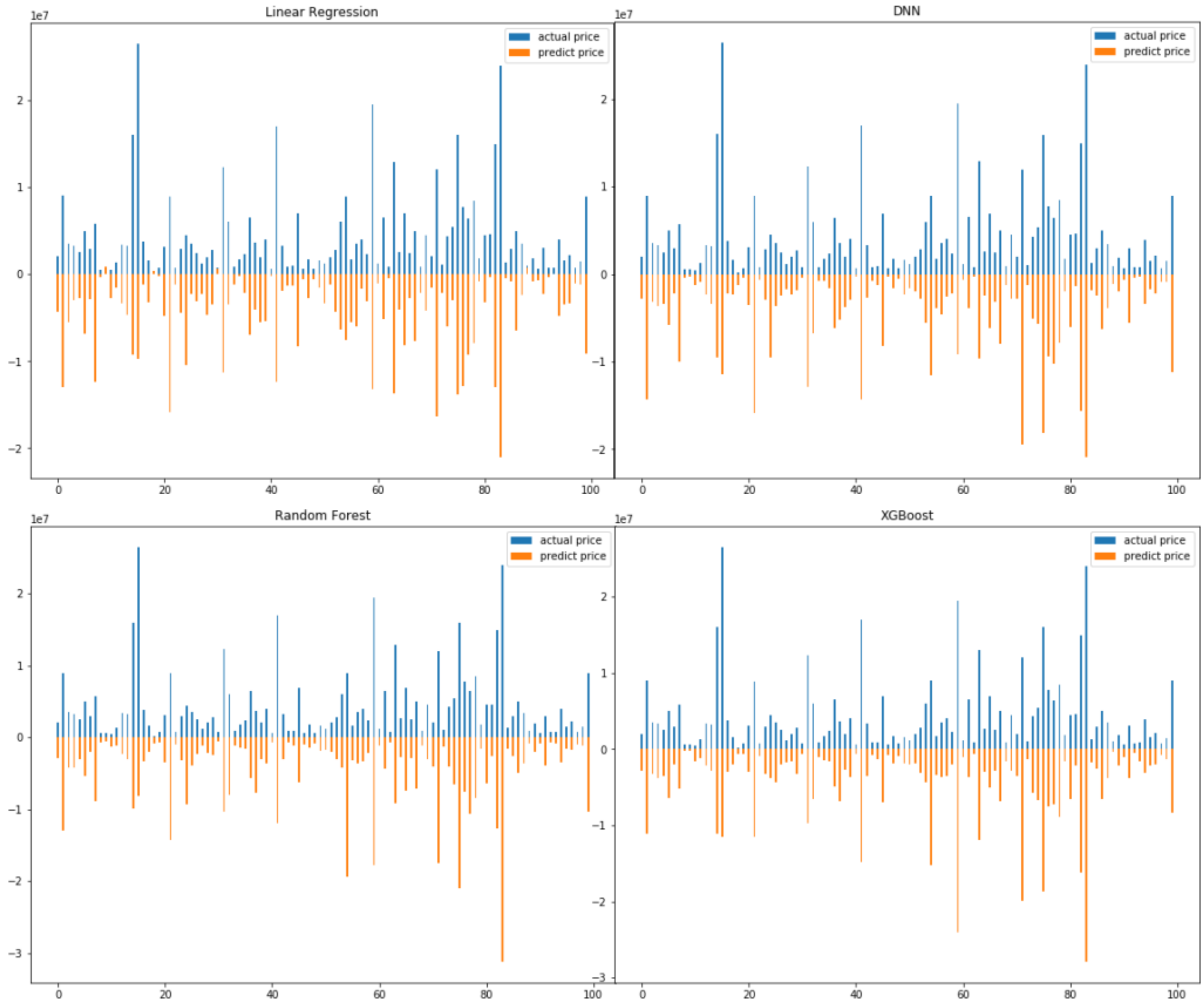


Fig. 6. Feature Importance

the real estate data and then we use restaurant price level near houses as the proxy for a neighborhood economic level and the distance between house and its nearest subway station as the proxy for house location. Based on data we obtained, we predict Manhattan sale house price using Linear Regression, DNN, RF and XGBoost. After testing models, we find that XGBoost and Random Forest have lower MSE compared with others.

4 Problems and Further Work

Firstly, based on the availability of the housing data, we can only obtain cross-sectional data with only one period. In the future, we can create a system for continually scraping sources for new information to track ongoing trends. Thus, more reasonable model like LSTM can be applied to analyze the house price. Secondly, we ignore the duplicate house information from different Real Estate websites in data processing. It is hard to match the addresses of the houses, since different websites has its own address format. In later research, more elaborate processing method can better clean the data. Thirdly, we have also scraped other useful features like the introduction and photos of the houses. We can further use NLP method to do sentiment analysis and CNN to do image segmentation in order to get more reliable features.

References

[1] Yang, Bowen Cao, Buyang. (2018). Ensemble Learning Based Housing Price Prediction Model.
[2] Feng, Yingyu Jones, Kelvyn. (2015). Comparing Multilevel Modelling and Artificial Neural Networks in House Price Prediction.
[3] Limsombunchai, Visit Gan, Christopher Lee, Minsoo. (2004). House Price Prediction: Hedonic Price Model vs. Artificial Neural Network. American Journal of Applied Sciences. 1. 10.3844/ajassp.2004.193.201.
[4] Chen, Xiaochen Wei, Lai Xu, Jiaxin. (2017). House Price Prediction Using LSTM.
[5] Yu, Li Anne et al. Prediction on Housing Price Based on Deep Learning. (2018).