House Price Prediction Using Neural Networks with Stacking Methods (Progress Report)

Team 158
Chris Armstrong (carmstrong37)
Haoyang Han (hhan77)
Jianxing Niu (jniu37)
Pengfei Mei (pmei8)
Shiyu Du (sdu37)
Weiqian Pan (wpan48)

(Word Count: 1879 excluding references)

1. Introduction - Motivation

This project is based on open-source housing price data provided by Zillow. During the project, our team will develop a complete pipeline for data cleaning, descriptive analysis, visualization, feature engineering, and modeling. Traditionally, statistical transformation and analysis would be used to perform housing price prediction. However, more modern techniques, such as machine learning, multi-task learning, and deep learning (LSTM), can also be used. This project will use an ensemble/stacking model combining traditional and contemporary techniques (LR, kNN, RF, XGBoost, and MLP) for predicting house prices.

2. Problem Definition

Housing price prediction plays a significant role in the modern real estate market.[8, 12,14] It is important for many private and public entities. Governments use house prices to determine interest rate policy, real estate buyers and sellers use prices to determine a "good" deal, and economic researchers use prices to judge the health of the economy. This project is to find a model (method) that can precisely predict house prices by combining multiple machine learning and deep learning algorithms.

We will develop methodologies from the following areas:

a. Machine learning algorithms, such as KNN, SVM, MLP, DT, etc;
b. Deep learning algorithms, such as ANN, Black Box model, LSTM, SRN, etc;
c. Linear Regression methods, such as SLR, GLM, Poisson Regression, etc.

3. Survey

Nowadays, many scholars research housing price prediction with various methods, like traditional time series models[3,13], classical machine learning algorithms[6, 12, 10], and classical deep learning algorithms[7,11, 17]. Researchers have also done a lot of work comparing different methods[6, 7, 17].

Reviewing all previous work, we have come to the following conclusions relevant to our project:

- Location-based classification (zone) is critical for housing price, which sets the baseline of the unit price[7]
- 81.7% of total variation in house price can be explained by living area, number of the bedrooms & bathrooms, lot size, and age of house[9]
- 4% of total variation in house price is affected by "soft conditions"[12], such as independent heating, good view, floor type, etc, which are not available in the dataset.
- There is no strong evidence for the role of macroeconomic fundamentals (unemployment rate, GDP growth, average income, etc) helping the model predict future home prices[8]
- Pure linear regression won't be accurate, more sophisticated machine learning or deep learning methods are required[5,7,11,12,10,13]

We also find limitations in previous work. For example, time series analysis requires a large training dataset for historical housing prices over a long time (~10 years or more), which is difficult to get. The DFM method seems to overestimate vibration at the price turning point. Some of the papers referenced for our proposal only apply a single method in each prediction without trying to combine multiple algorithms to make a better prediction.

4. Proposed Algorithm Design and Implementation

In this section, we will talk about the innovation part of the project, what we hope to achieve with our novel approach, and our implementation.

4.1 Innovation

Deep Learning has become a buzzword over the past few years, with encouraging results from object detection & recognition, speech recognition, and even generating fake images or fake news (OpenAI's GPT2). These accomplishments clearly demonstrate the capabilities of deep learning technologies. Encouraged by this, we will attempt to apply deep learning to this project.

We are trying to improve upon the previous work in various ways:

4.1.1 The first approach is innovative in the sense that we are applying techniques from Convolutional Neural Networks (CNN) to traditional Deep Neural Networks.

- We are employing various activation functions (sigmoid & ReLU) and a dropout layer.
- We will try the idea of skip-connection from the classic CNN model, aka ResNet: allowing neurons in a lower layer to skip a few layers and connect to a much higher layer. The rational is that usually the deeper the network is, the more capable it is to learn a good function to solve the problem. However, if there are too many layers, the parameters become too hard to tune, so by adding residual connections, learning deeper models becomes feasible. The schema for residual connection is shown below[19].

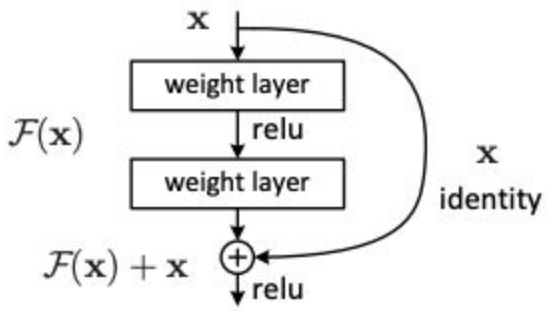


Figure 2. Residual learning: a building block.

- We will compare different optimizers, such as RMSProp, Adagrad, Adam, and Nadam.
- We believe we can achieve better performance with these innovations, and by also combining multiple neural networks or combining the best neural networks with other models, such as SVM.

4.1.2 The second approach is to develop a stacking algorithm, with kNN, XGBoost, LR, RF, and MLP in one model. Stacking is trying to utilize the advantages of multiple models by combining their performance together via a meta-regressor (in this case, logistic regression or xgboost regressor). The base models are trained on the same training dataset while the meta-model uses the outputs of the base model as features to avoid overfitting. It's quite normal to

combine several machine learning algorithms together using stacking, but very few people would combine deep learning models into the stacking layer. Since machine learning model-based stacking would improve the performance of prediction, we believe that adding deep learning models could have a similar effect.

4.2 Implementation (Under Construction)

We used the keras framework to build neural networks, and we are trying the various aforementioned techniques. So far we have trained two-layer neural networks and they already shows promising results.

Regarding the stacking method, Sklearn, Xgboost, and vecstack packages were used in the implementation. During the initial attempt, we used logistic regression, decision tree, random forest, multi-layer perceptron, k-nearest neighbor, and xgboost for ensemble. This is further explained in the next section.

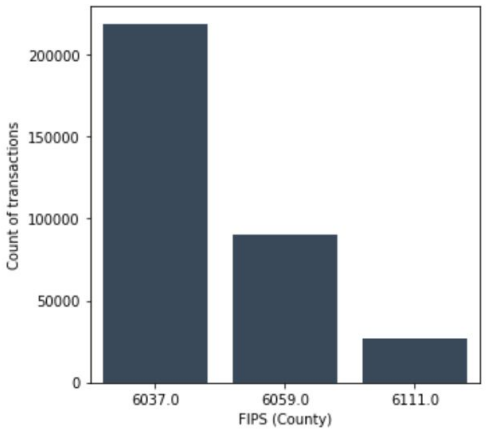
5. Experiments & Evaluation

The general idea of this experiment is to design a standard procedure for data pre-processing to make sure all models can be fed with the same dataset. There are usually several steps: data exploration and sampling, data cleaning, feature scaling and normalization, and feature selection, etc. Then we make the model fitting by linear regression as a baseline. Furthermore, we will compare different machine learning/deep learning models' performance. Finally, we use the stacking method to combine multiple models together.

5.1 Data preprocessing

We designed a standard pipeline of data preprocessing in our project.

- Importing data. We imported data which were downloaded directly from https://www.kaggle.com/c/zillow-prize-1/data and merged into a single set.
- Data exploration and sampling. We analyzed the distribution of transaction dates and property location. Below is the Distribution of property location:



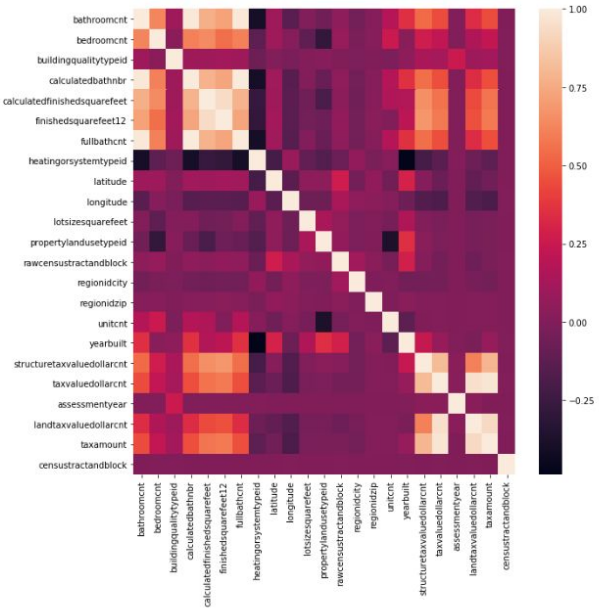
Because there are more samples in Los Angeles (FIPS: 6037), we chose data of samples in Los Angeles as a subset of data.

- Data cleaning. We dropped all features which have >= 10% missing values. Also, we use 0 to replace NaN values.
- Feature scaling and normalization. We use standardization to scale and normalize features data.
- Feature selection. We ranked features' importance using extra trees and dropped the feature with extremely low importance (1.98685e-08). Below is the ranking of feature importance:

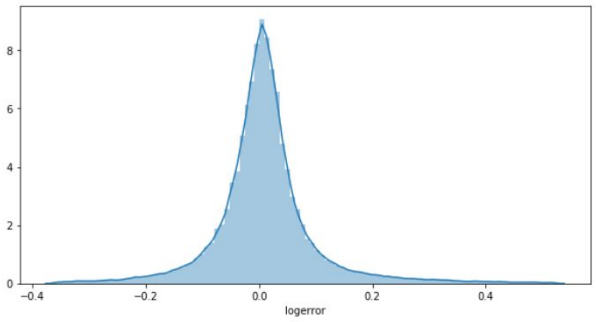
Out[15]:

	importance
lotsizesquarefeet	0.0872329
taxamount	0.085145
yearbuilt	0.0764289
calculatedfinishedsquarefeet	0.0715175
structuretaxvaluedollarcnt	0.0709051
latitude	0.0685489
longitude	0.0661421
landtaxvaluedollarcnt	0.064507
finishedsquarefeet12	0.0628053
taxvaluedollarcnt	0.056028
bedroomcnt	0.0456951
regionidzip	0.0451812
censustractandblock	0.0425625
rawcensustractandblock	0.0420605
regionidcity	0.0273993
heatingorsystemtypeid	0.0188105
propertylandusetypeid	0.0142424
bathroomcnt	0.0124015
calculatedbathnbr	0.0113157
unitcnt	0.0107257
fullbathcnt	0.00981202
buildingqualitytypeid	0.00933361
assessmentyear	0.00119925
roomcnt	1.98685e-08

Then we analyzed feature correlation using a heatmap and dropped one of two features that have a correlation higher than 0.8. Below is the heatmap of the features correlation:

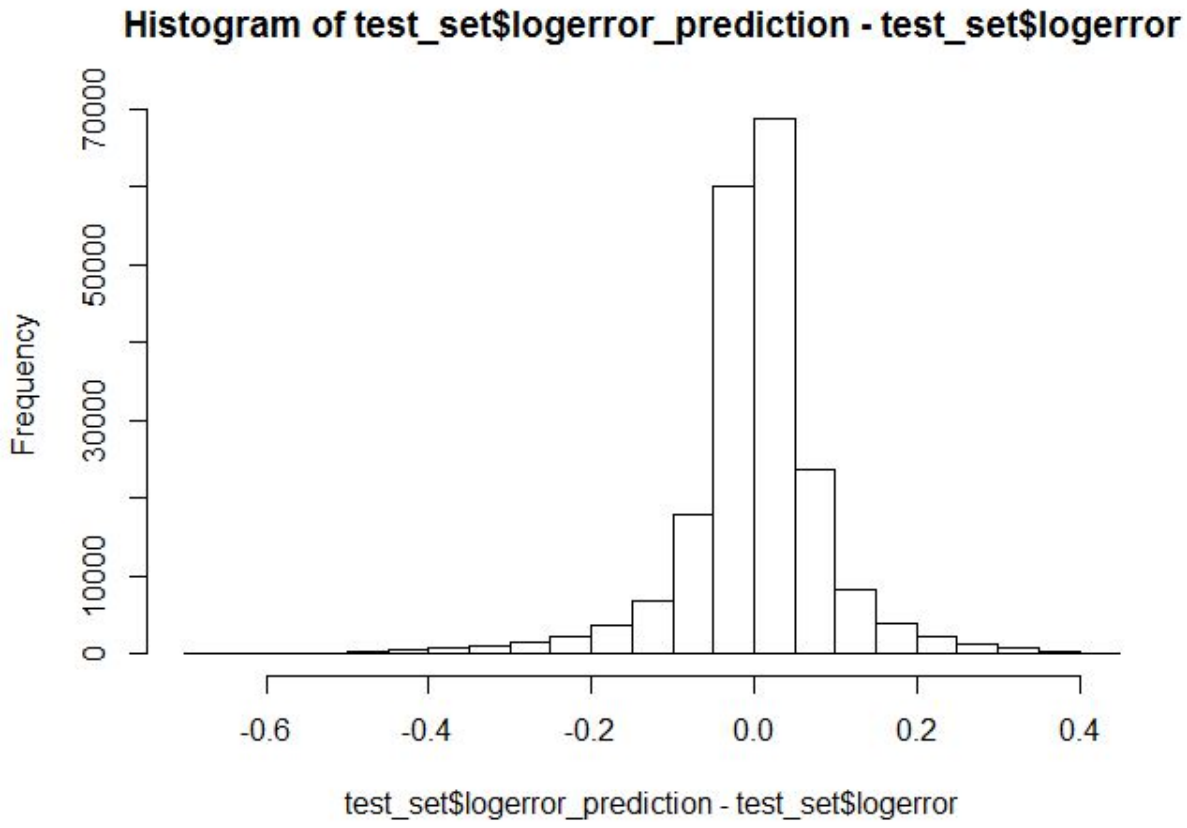


- Removing outliers. Finally, we found there is a nice normal distribution over the target (logerror). At the same time, we dropped the outliers of logerror (> 99 or < 1). Below is the normal distribution of logerror:

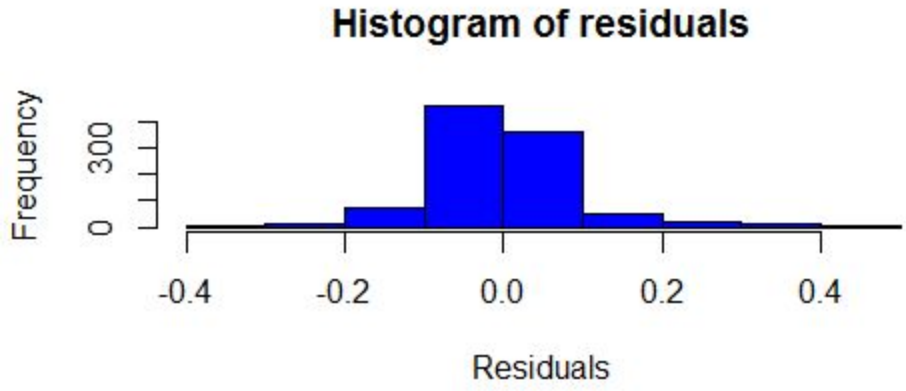


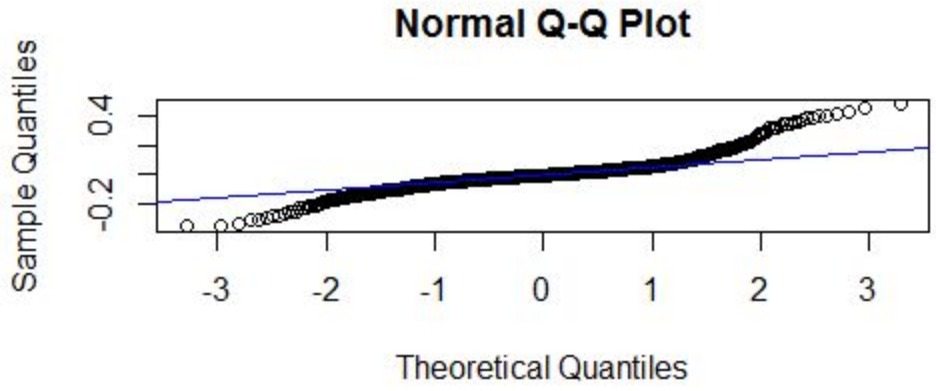
5.2 Benchmark for house price prediction

The multivariate linear regression (MLR) model showed decent accuracy as shown in the histogram of prediction error below.

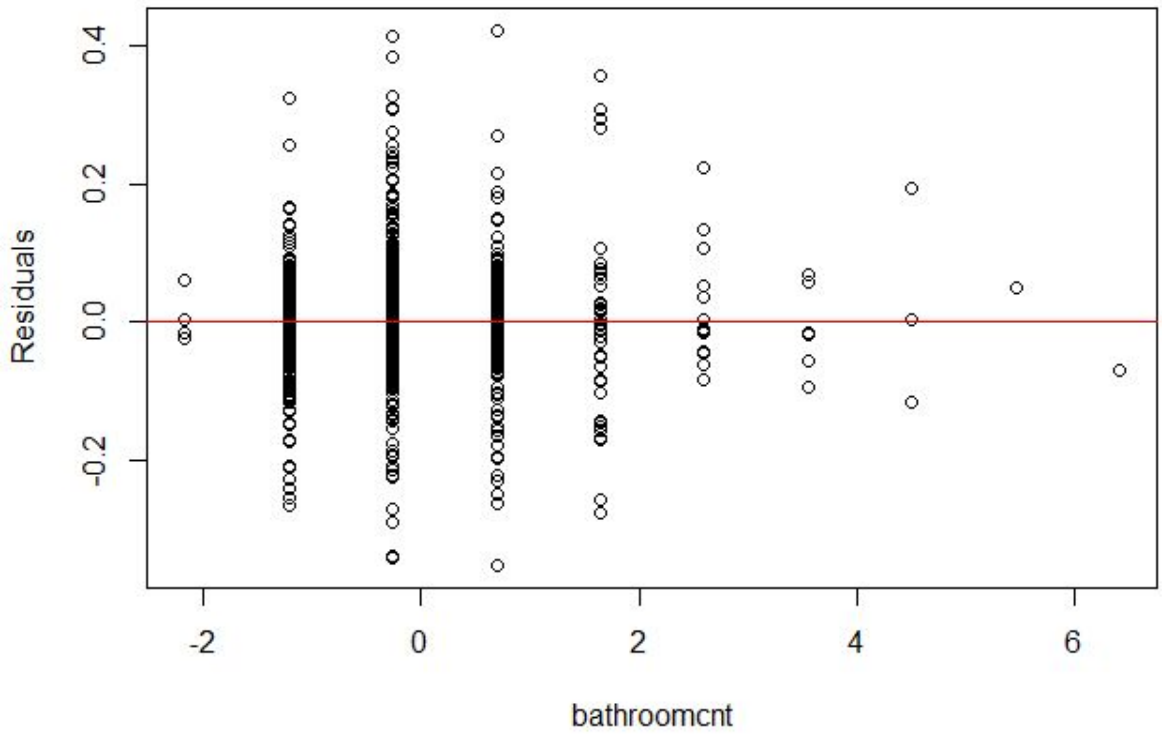


However, the model QQ-plot shows rather severe departure from normality, possibly indicating the need for a preliminary data transformation.





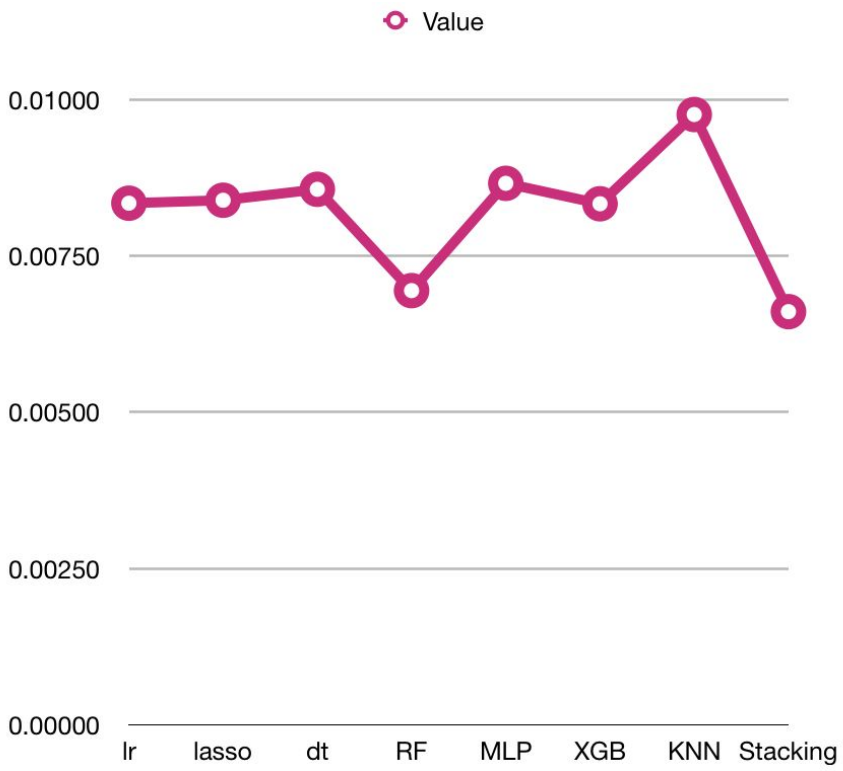
The model also did not properly treat categorical variables (such as quantitative variables with low numbers of values). Thus, in the example residual vs. predictor chart, we see inappropriate behavior in the form of “leveled” residual clusters (predictor/residual discretization).



Overall we believe that a linear regression model is appropriate for our data and the shortcomings of our initial model will be address in the final model.

5.3 Performance of all other machine learning models.

We finished the baseline fitting with MSE and evaluation metrics and below is the result of the different models:



It's clear to see that stacking model had a better performance compared to any other single models in this experiment.

5.4 Performance of deep learning models.

After comparing different machine learning algorithms, we know exactly where the baseline is. So we should build several different deep learning models with different activation functions, layers, and node sizes.

5.5 Parameter Tuning.

Here we will use a grid search looking for optimized hyper-parameters for each model. Better base models will improve our final model.

6. Plan of Activities

Our future work will be focusing on refining our linear regression, exploring deep learning models, like ANN, Black Box model, LSTM, SRN, then utilizing STACK techniques to improve overall performance.
All team members have contributed equally. Our proposed and actual schedules are as follows.

Previous Proposed Project Schedule		
Date	**Content**	**By**
March 4	Complete proposal	Everyone
March 4	Complete proposal presentation	Chris Armstrong
March 7	Gather dataset	Everyone
March 14	Data pre-processing and statistical analysis	Weiqian Pan
March 21	Develop model fitting pipeline and benchmarks	Everyone
March 29	Progress Report	Everyone
April 5	Develop multiple models for comparison	Jianxing Niu, Haoyang Han, Pengfei Mei
April 12	Grid search/parameter tuning	Shiyu Du
April 15	Draw conclusions and generate report	Chris Armstrong
April 19	Final Report	Everyone
April 19	Final Presentation	Everyone

Revised Proposed Project Schedule		
Date	**Content**	**By**
March 4	Complete proposal	Everyone
March 4	Complete proposal presentation	Chris Armstrong

March 10	Gather dataset	Everyone
March 14	Data pre-processing and statistical analysis	Weiqian Pan and Haoyang Han
March 21	Develop model fitting pipeline and benchmarks	Haoyang Han
March 30	Progress Report	Everyone
April 9	Develop multiple models for comparison (Try different optimizers, number of layers, and skip-connections)	Jianxing Niu, Haoyang Han, Pengfei Mei
April 13	Grid search/parameter tuning	Shiyu Du
April 15	Draw conclusions and generate report	Chris Armstrong
April 19	Final Report	Everyone
April 19	Final Presentation	Everyone

7. Conclusion & Discussion

We have implemented multiple data cleaning methods, like outlier detection, scaling, normalization, feature selection, etc, on the raw dataset provided by Kaggle (Zillow Prized competition), and picked 214080 real estate transaction records in Los Angeles County as the dataset to build housing price models. We then randomly split the dataset into the training set (66%, 143433 transactions) and test set (33%, 70647 transactions). We chose 17 predicting variables based on variables' importance analysis, making sure the model had a strong predicting power. We have implemented 7 models: linear regression, LASSO regression, decision tree (DT) regression, random forest tree (RFT) regression, Multi-layer Perceptron (MLP) regression, Extreme Gradient Boosting (XGB) regression, and KNN regression. We adopted the mean square error (MSE) method to verify and compare predictions. RFT regression performs best, with MSE equal to 0.007. We generated an ensemble model using all above base models and trained & tested. The stacking model performs best, with MSE

equal to 0.0065. This proves our innovation hypothesis that stacking methods will improve price prediction accuracy.

8. References

[1] Baldominos A., Blanco I., Moreno A. J., Iturrarte R., Bernárdez Ó., Afonso C. (2018 Nov 21). Identifying Real Estate Opportunities Using Machine Learning. Retrieved from https://arXiv:1809.04933v2

[2] Bency, A. J., Rallapalli, S., Ganti, R. K., Srivatsa, M., Manjunath, B. S. (2017). Beyond Spatial Auto-Regressive Models: Predicting Housing Prices with Satellite Imagery. 2017 IEEE Winter Conference on Applications of Computer Vision (WACV). doi: 10.1109/wacv.2017.42

[3] Chen X., Wei L., Xu J. (2017 Sep 25). House price prediction using LSTM. Retrieved from https://arXiv:1709.08432

[4] Choudhary, P., Jain, A., & Baijal, R. (2018). Unravelling Airbnb Predicting Price for New Listing. Retrieved from https://arxiv.org/pdf/1805.12101.pdf

[5] Fan C., Cui Z., Zhong X. (2018 Feb) House Prices Prediction with Machine Learning Algorithms. Retrieved from http://delivery.acm.org/10.1145/3200000/3195133/p6-Fan.pdf?ip=123.126.70.237&id=3195133&acc=ACTIVE%20SERVICE&key=A79D83B43E50B5B8%2E5E2401E94B5C98E0%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&__acm__=1551414804_28fb84672ad92bbdbc8277634f1421f4

[6] Gao G., Bao Z., Cao J., Qin A. K., Sellis T., Wu Z. (2019 Jan 7). Location-Centered House Price Prediction: A Multi-Task Learning Approach. Retrieved from https://arxiv.org/abs/1901.01774

[7] Goodman A. C., Thibodeau T. G. (2003 Sep). Housing market segmentation and hedonic prediction accuracy. Retrieved from https://www.sciencedirect.com/science/article/pii/S1051137703000317

[8] Gupta R., Kabundi A., Miller S. M. (2009 May). Using Large Data Sets to Forecast House Prices: A Case Study of Twenty U.S. States. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.464.2380&rep=rep1&type=pdf

[9] Khamis A. B., Kamarudin N. K. B. (2014 Dec). Comparative Study On Estimate House Price Using Statistical And Neural Network Model. Retrieved from http://www.ijstr.org/final-print/dec2014/Comparative-Study-On-Estimate-House-Price-Using-Statistical-And-Neural-Network-Model-.pdf

[10] Li Y. (2011 July) Forecasting Housing Prices: Dynamic Factor Model versus LBVAR Model. Retrieved from

https://ageconsearch.umn.edu/record/103667/files/AAEA-Forecasting%20Housing%20Prices.pdf
[11] Limsombunchai V., Gan C., Lee M. (2004 March). House Price Prediction: Hedonic Price Model vs. Artificial Neural Network. https://researcharchive.lincoln.ac.nz/bitstream/handle/10182/5198/House_%20price_%20prediction.pdf;sequence=1
[12] Morano P., Tajani F., Torre C. M. (2015 Jan). Artificial Intelligence in Property Valuations. An Application of Artificial Neural Networks to Housing Appraisal. Retrieved from https://pdfs.semanticscholar.org/57de/46ad59391c6b120d076863e5387111a60a1d.pdf
[13] Mu J., Wu F., Zhang A. (2014 Aug 4). Housing Value Forecasting Based on Machine Learning Methods. Retrieved from https://www.hindawi.com/journals/aaa/2014/648047/
[14] Muzumdar, P. (2014). Effects Of Zoning On Housing Option Value. Journal of Business & Economics Research (JBER), 9(5), 41. doi: 10.19030/jber.v9i5.9026
[15] Nagaraja C. H., Brown L. D., Zhao L. H. (2011 Apr 14). An autoregressive approach to house price modeling. Retrieved from arXiv:1104.2719
[16] Oxenstierna, J. (2017). Predicting house prices using Ensemble Learning with Cluster Aggregations (Dissertation). Retrieved from http://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-345157
[17] Park B., Bae J. K. (2015 April 15). Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data. Retrieved from https://www.sciencedirect.com/science/article/pii/S0957417414007325
[18] Phan T. D. (2018 Dec). Housing Price Prediction Using Machine Learning Algorithms: The Case of Melbourne City, Australia. Retrieved from https://ieeexplore-ieee-org.prx.library.gatech.edu/stamp/stamp.jsp?tp=&arnumber=8614000
[19] Kaiming He (2016). Deep Residual Learning for Image Recognition. Retrieved from https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper.pdf