San Diego House Price Predicting

**Wen Yan, Mengting Wang, Salah Ahmad, Xia Song**

### Hypothesis Definition

In a specific city, area or neighborhood, house price is mainly determined by some house features, for example, square feet, number of bedrooms, number of bathroom, house age, house view and number of garage, etc al. It means house price is predictable. In a community, according to house features and historical sale data, we can estimate one house’s price.

### Analytic Approach for MVP

### All possible inputs:

**Sqft**: square feet, continuous data type.

**Num\_bed**: number of bedroom, continuous data type.

**Num\_bath**: number of bathroom, continuous data type.

**View**: house with view or not, ordinal Categories (1 represent with view, 0 represent none).  
**Pool**: house with pool or not, ordinal Categories (1 represent with view, 0 represent none).  
**Sqft\_zip\_avg**: average square feet in zip code level, continuous data type.

**Sqft\_price\_zip\_avg**: average square feet price in zip level, continuous data type.

**Sold\_price\_zip\_avg**: average sold price in zip level, continuous data type.

### Targets:

**Log of sold\_price**: natural logarithm of sold price.

When we check the distribution of the target variable, we found the distribution of sold price is right skew. One assumption of linear regression models is that the error between the observed and predicted (i.e., root-mean-square error) should be normally distributed. Violations of this assumption often stem from a skewed response variable.

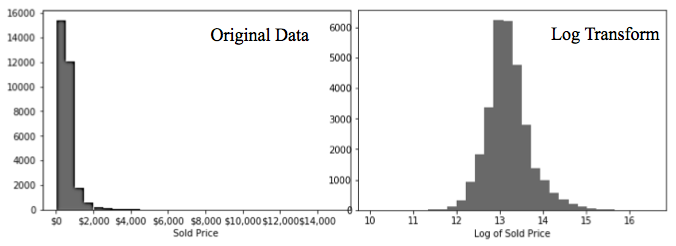


Fig 1. Distribution of original sold price and natural logarithm transformed sold price

**Criteria**: mean square error. The MSE is a measure of the quality of an estimator—it is always non-negative, and values closer to zero are better. natural logarithm of sold price.

### Models:

**Train and test data**

Currently, we are still working on data feature engineering, more features will be provided. During this test period, one year data (oct 2016 to oct 2017) were used to train, validate and test our models. For one year, there are totally 37929 records, and 80% of data records (30343 records) were used as train data and 20% of data records (7586 records) were used as test data.

**Model validation**

Since there is only one year for our models, cross-validation was used to assess how the model will predict in a independent data set.

**Adopted models**

Three different models are adopted to predict house sold price in San Diego area. The information in the table 1 represents our preliminary results for each model. The table explains the pros and cons for each model type, the optimal hyperparameters found through either grid search or Bayesian optimization, our test score. Our scores the mean square error (MSE) of our predictions, which is a metric for describing the difference between the observed values and our predicted values for sold price; scores closer to zero are better.

Table 1. Models and models’ parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Models | Pros | Cons | Parameters | Cross Validated MSE Score |
| Random Forest | Lower variance, Correlated data, scale invariant | High bias, difficult to interpret | N\_estimator = 1000  N\_depth = 7  N\_features = 8 | 0.072 |
| Gradient Boost | Feature scaling not needed high accuracy | Expensive computation, overfitting | N\_estimator = 350  N\_depth = 6  N\_features = 2  Leaf = 10  Learning rate = 0.05 | 0.25 |
| Linear Regression | Easily interpretable, inexpensively computation, less prone to overfitting | Required scaled variable, require numerical data set | Alpha = 0.5 | 0.30 |

**Importance of features**

Except predicting, random forest and gradient boost can also provide very straightforward method for feature ranking.

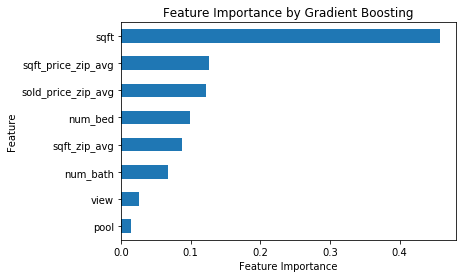
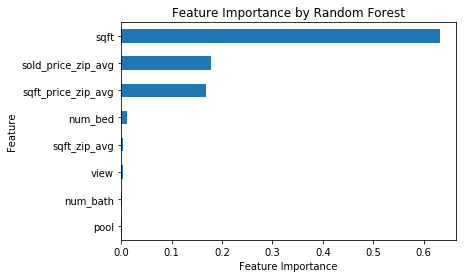


Fig 2. Feature importance evaluated by Random Forest and Gradient Boosting

Both Random Forest and Gradient Boost models rank square feet as the most important feature, and sold price and square feet price in zip code level are two second important features. In addition number of bedrooms, square feet in zip code level and number of bathrooms are also obviously contributed to Gradient Boosting model simulation however, these features are not significantly contribute to Random Forest model simulation (Fig 2).

**Results of model and model performance**

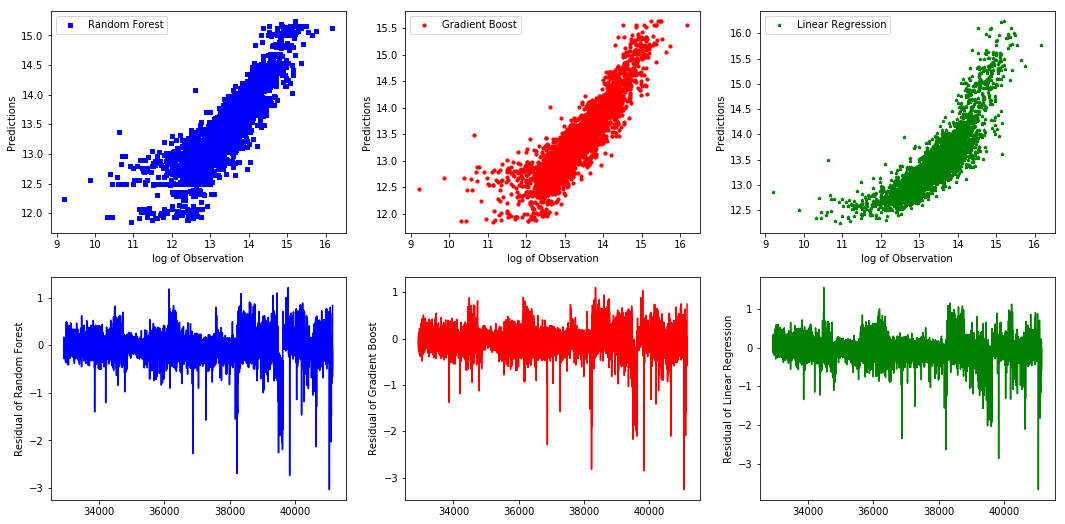


Fig 3. Feature importance evaluated by Random Forest and Gradient Boosting

The objective of this report was to build models to predict housing prices of different residences in San Diego area, CA. According to MSE score of three models, 0.072 for Random Forest, 0.25 for Gradient Boosting and 0.30 for linear regression respectively (Table 1), Random Forest model performed best in predicting house price. In addition, Fig 3 demonstrated the comparison between observed values and predicted values and residuals of each model. From the comparison and residual pattern, we also can conclude that tree based models (Random Forest and Gradient Boosting) are also performed better than linear model in predicting house price.

**Next Steps of modeling**

1. More features planned:

lot\_size: lot size in sqft

first\_floor\_area: first floor size in sqft

sold\_age: the year count from year built to year sold

distance\_to\_primary/middle/high\_school: depending on school boundary GIS data

score\_of\_primary/middle/high\_school: depending on school boundary GIS

sqft\_bracket: categorical feature that which sqft bracket the property belongs to

geo\_grid\_bracket: instead of zip/city, we’ll try to create our own geo grids and then cluster them to get categorical feature for each property

1. More data from previous years

Beyond last year sales datasets, we will try using more years’ data to see how the model will perform. However, it’s known that the more early the data is used, the more model will be affected by the time factor.

1. Segmentation on key features and multiple models

Instead of training one general model, we’ll try clustering on single key feature or various combination of features, and train separate model for each cluster. Key features include property\_type, sqft, geo\_grid, year\_built.

1. Modeling pipeline

ModelDB will be deployed to record each modeling parameters, dataset features used and metrics score. And wrapper python modules will be developed to encapsulate the dataset loading, dataset dividing for training/validation/testing, using various models with hyperparameters and syncing to ModelDB.

**Beyond modeling**

Hopefully if the modeling of price evaluation has good performance, we can train models for each year and evaluate the price for all properties for all years. Then with this filled complete time-series data, we can research on the short/long-term trending for property value, cluster the properties based on their trending and study the correlation of the trending with various property features.

**Bullets for each team member’s individual contribution in step 4**

Wen: Implemented the wrapper class for data source connection and data preprocessing, so the team can get exact same dataset by calling same function. And worked on feature engineering for non-direct features.  
Mengting: Continue to work on data explorations; perform visualizations; monitoring overall team progress and managing resource allocations among team members; producing PowerPoint slides for group presentation  
Salah: Continue to work on getting the new features by reaching out to different groups and entities; gain further clarification on existing county data  
Xia: Design and set up models. Write report and wrap up model performance in predicting house price.

**Major updates comparing to step 1 through 3**

Key features in San Diego County house data are ready to use

Made major improvements in data cleaning and feature engineering; produced couple of key features for baseline model prediction

Set up different machine learning models as baseline to predict San Diego house price.