Machine Learning Final Project

Zillow's Home Value Prediction (Zestimate)

Sinuo Xu Hairun Wang

Business Problem

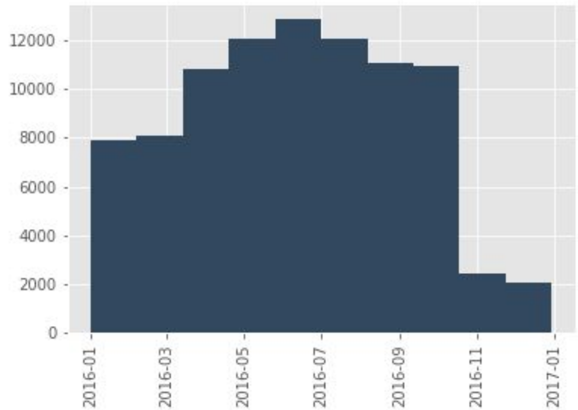
- Zillow is an online real estate database with data on homes across the United States.
- One of Zillow's most popular features is a proprietary property value prediction algorithm: the Zestimate.
- This feature is a hot-button topic across property sellers, buyers and agents.

Objective

- Zillow is constantly trying to improve its Zestimate.
- The objective is to help advance Zestimate accuracy even further by predicting the logerror between the Zestimate and the actual sales price of home. The log error is defined as:
  - *Logerror=log(Zestimate)−log(SalePrice)*
- Model performance is evaluated on Mean Absolute Error between the predicted log error and the actual log error

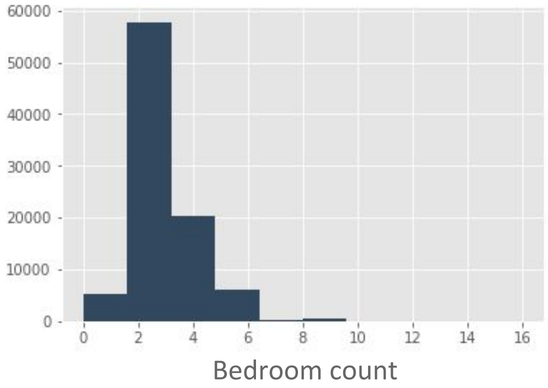
Dataset Description

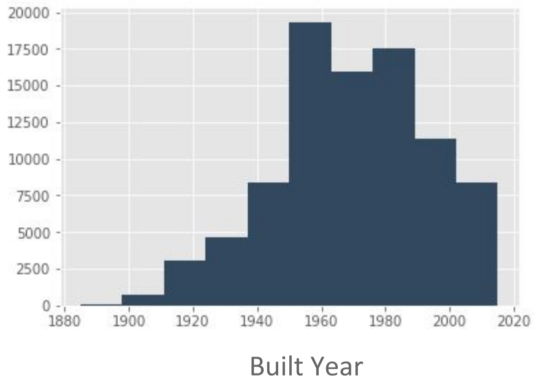
- Properties transactions data in three counties (Los Angeles, Orange and Ventura, California) data in 2016
- The original dataset has **~3M** transactions and **58** features such as built year, bedroom count, sqrt, zip code and etc.

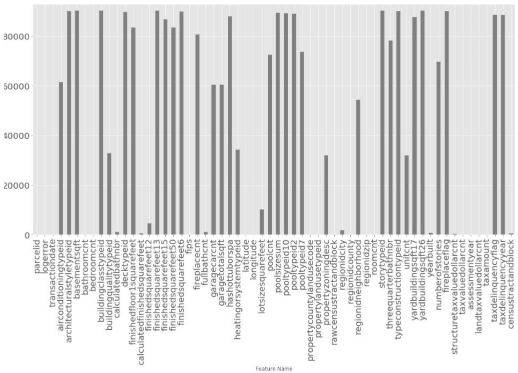


fips
fireplacecnt
fullbathcnt
garagecarcnt
garagetotalsqft
hashottuborspa
heatingorsystemtypeid
latitude
longitude
lotsizesquarefeet
poolcnt
poolsizesum
pooltypeid10
pooltypeid2
pooltypeid7
propertycountylandusecode
propertylandusetypeid
propertyzoningdesc
rawcensustractandblock
regionidcity
regionidcounty
regionidneighborhood
regionidzip
roomcnt
storytypeid
threequarterbathnbr

Exploratory Data Analysis







- > 90% Missing values for most of the features
- ~ 124 properties have more than one sales in 2016
- Assumptions: Model output (logerror) is driven by features like bedroom count, built year, location etc

Data Preprocessing

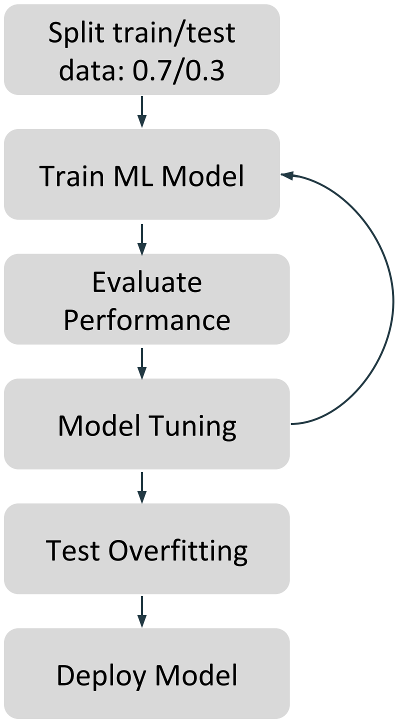
Duplicate	Missing Value	Data Format	Categorical Data	Scaling
Removed duplicates (124 properties) which had more than one transactions in the year of 2016	Kept features which have more than 80% not null values. Filled these empties with the median value. **33** features left after the imputation.	Re-formatted Datetime Columns and extracted the most useful transaction month of it because all transactions happened in the 2016.	Convert categorical data into dummy variables using get_dummies	For the linear regression and SVR, scaled the dataset with the Standardscaler().

After data cleaning process, we have 90K transactions

Approach

Predicting log error is a regression problem since outputs are continuous values. Due to the fact that dataset is highly dimensional, the following ML models are implemented:

- Linear Regression (Baseline model)
- Lasso Regression
- Support Vector Regressor
- Random Forest



Linear & Lasso Regression - Baseline

The Linear Regression is selected as our baseline model at first. After the basic Linear Regression, we tried to optimize the performance of the regression model. Considering we have more than 30 features in the data which may be too many to explain, Lasso Linear Regression is good for feature selection and regularization.

Top 5 features of the Linear Regression:

1. Finished square feet
2. Garage total square feet
3. Tax amount paid
4. Calculated finished square feet
5. Region id county

Linear & Lasso Regression: Performance Evaluation

Linear Regression	Lasso Regression
Test MAE: 0.06852	Test MAE: 0.06824
Train MAE: 0.06828	Train MAE: 0.06896
	No. of features used: 11

Based on the train MAE and the test MAE. Two models don’t have overfitting issue.

SVR with Randomized Search

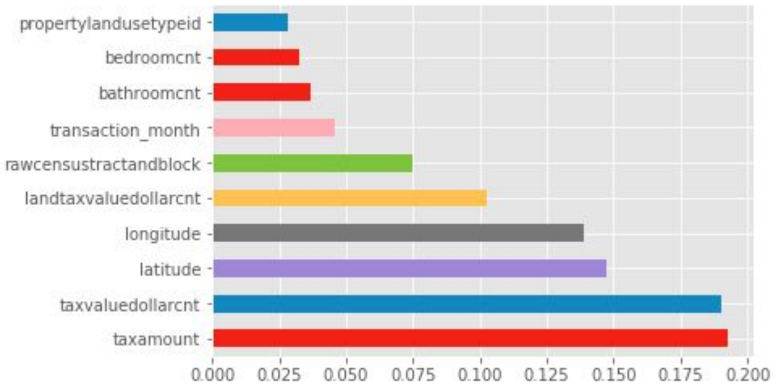
- **Motivation:** The method of Support Vector Classification can be extended to solve regression problems. This method is called Support Vector Regression.
- **Hyperparameter tuning:** Randomized Search
- No overfitting found

Parameters	MAE
C=1.0, cache_size=200, degree=3, epsilon=0.1, kernel=’rbf’, max_iter=-1, shrinking=True	0.07105
C=2.1, cache_size=200, degree=3, epsilon=0.1, kernel=’rbf’, max_iter=-1, shrinking=True	0.07162
Predictions on train data to test signs of overfitting	0.06935

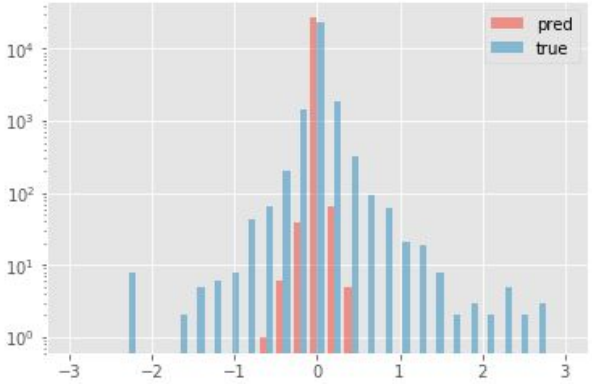
Random Forest

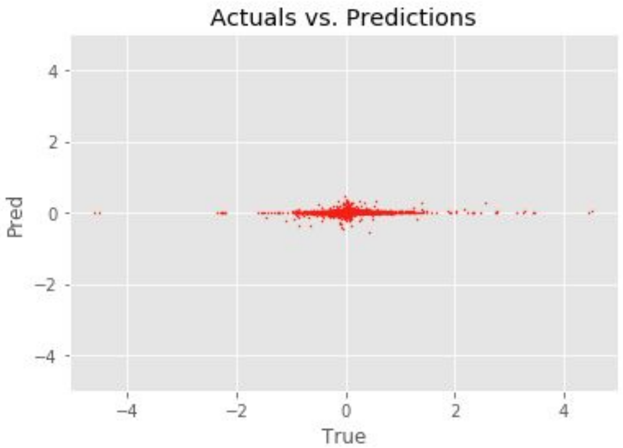
- **Motivation:** Great with high dimensional data; Quick prediction/training Speed; No need to rescale or transform the data
- **Hyperparameter tuning:** Randomized Search
- **Top 5 features** are ‘taxamount’, ‘taxvaluedollarcnt’, ‘latitude’, ‘longitude’, ‘landtaxvaluecnt’
- No overfitting found

Parameters	MAE
n_estimators=100,max_depth=9,min_samples_split=6,min_samples_leaf=4, random_state = 42	0.06867
n_estimators=10,max_depth=5,max_features=7,min_samples_split=5,min_samples_leaf=7,random_state=42	0.06849
Predictions on train data to test signs of overfitting	0.06814



Random Forest (Cont.)





- True values span between -4 to 4, but predictions only lie between -1 to 1
- Complexity of the model
- Features in the dataset do not explain random human behaviors

Conclusion & Future Work

Model Name	Mean Absolute Error
Lasso Regression	0.06824
Random Forest - Randomized Search	0.06849
SVR	0.07105

- Final model selection based on the lowest MAE - **Lasso Regression**
- Tax amount, location, square feet are most important features to predict the value of a property.
- Conduct analysis on impact of macroeconomics factors to home value prediction, such as interest rate, inflation rate
- Increase the depth of dataset: add data from other states and data from 2018, 2019

Thank You