Predicting Zillow Estimation Error Using Linear Regression and Gradient Boosting
Darshan Sangani, Kelby Erickson, and Mohammad Al Hasan

Introduction
As the leading real estate database company, Zillow has developed the most accurate property value forecasting algorithm to date. However, like any prediction mechanism, Zillow’s algorithm is not perfect. Seeking to improve upon Zillow’s current prediction mechanism, we tested the effectiveness of several different machine learning models and techniques (described in Methods) at making property related forecasts.

Data Set
We had an input training matrix with different properties along the vertical axis and different property features (e.g., number of bedrooms, square footage, etc.) along the horizontal axis. For each of these properties, we had a corresponding output, the log error. This is defined as the difference between the log of Zillow’s estimate of a property’s sale price and the log of the actual sale price:

\[ \text{logerror} = \log(\text{Estimate}) - \log(\text{SalePrice}) \]

Methods
• Linear Regression: machine learning algorithm that uses a least squares approach to find a line of best fit that accurately models the data
• Gradient Boosting: an algorithm that works on the principle of ensemble learning, in which an ensemble of decision trees is built and the predictions of individual trees are summed in order to calculate the overall prediction

One parameter of the gradient boosting algorithm is the loss function that the algorithm seeks to minimize while building trees. The loss functions that we used are least squares (LS) and least absolute deviation (LAD). LS is the sum of squared differences between predicted and actual values while LAD is the sum of absolute differences between predicted and actual values.
• XGBoost: extreme gradient boosting is an optimized version of gradient boosting
• Grid Search: the gradient boosting algorithm has a number of parameters that need to be finely tuned in order to develop an accurate prediction model; in order to accomplish this, we used a grid search algorithm, which determines the best combination of parameter values to input to the gradient boosting algorithm
• Normalization: adjusts input training matrix so that all property features have the same scaling
• Principal Component Analysis (PCA): reduces the number of features in the input training matrix by combining features that are strongly correlated
• Time as a property feature: unlike previous research, we added a month feature to our input matrix in order to predict values for different months; in other words, we used time like we would any other property feature

Evaluation Criterion
In order to measure how accurately a model makes predictions, we calculated mean absolute error (MAE), which is the average absolute difference between predicted values and actual values. The lower the MAE, the more accurately the model makes forecasts.

<table>
<thead>
<tr>
<th>Model</th>
<th>MAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>LR</td>
<td>0.0131</td>
</tr>
<tr>
<td>LR (with normalization)</td>
<td>0.0129</td>
</tr>
<tr>
<td>GB (with LS loss)</td>
<td>0.0052</td>
</tr>
<tr>
<td>GB (with LAD loss)</td>
<td>0.0050</td>
</tr>
<tr>
<td>GB (with LAD loss &amp; PCA)</td>
<td>0.0048</td>
</tr>
<tr>
<td>GB (with LAD loss &amp; grid search)</td>
<td>0.0044</td>
</tr>
</tbody>
</table>

Conclusion
• Gradient boosting outperforms linear regression
• Using an LAD loss function works better than using an LS loss function
• Gradient boosting combined with grid search parameter optimization resulted in the best prediction model

This research was made possible with the support of the Indiana University-Purdue University Indianapolis Department of Computer Information and Information Science, as well as through funding from the National Science Foundation and the United States Department of Defense. The authors would like to thank their mentor Dr. Mohammad Al Hasan, as well as Dr. Feng Li, Dr. Eugenia Fernandez, and Sheila Walter for their support.