Module 3 Final Project:

Modeling Tax-Assessed Home Values Using Zillow Housing Data

DX603 Spring 2025 - Team 37

- Zonglin Wu
- Sergey Nelyapenko
- Lee McFarling

Project Overview:

Project Objectives:

- Develop a predictive model using Zillow's housing dataset to estimate the tax-assessed value of residential properties (taxvaluedollarcnt).
    - Identify key factors influencing property valuations.
    - Minimize prediction errors using Root Mean Squared Error (RMSE).

Approach:

- Leverage Zillow's Dataset
    - Expand on Zillow's Zestimate tool by predicting tax-assessed values

Key Findings:

- **Random Forest**
    - Top performer on test set
    - Top performer on cross-validated (CV) RMSE
- **Gradient Boosted Trees**
    - Second best performer in CV RMSE
    - Performed worse than both Random Forest and Linear Regression on test set
    - *Potential overfitting?*
- **Linear Regression**
    - Outperformed Ridge and Lasso regression models.
    - Feature engineering process minimized multicollinearity and feature noise. Also well regularized.

Introduction:

Zillow’s Zestimate Impact:

- Zestimate revolutionized real estate transparency
    - Provides instant, market-based property valuations.
    - Enabled consumers to easily compare listing prices to market estimates with a single click.
    - Empowered informed decision making when evaluating real estate prices

The Overlooked Metric — Tax-Assessed Value:

- Property taxes remain a significant and ongoing expense for homeowners.
- Home Tax Value Assessment remains shrouded in opacity.
    - Difficult for consumers to evaluate or predict

Project Purpose:

- Develop a machine learning model to accurately predict tax-assessed values using Zillow’s housing database.
- Bring greater transparency to an often overlooked but (financially) critical metric

Dataset Description

Dataset Source

- Zillow's 2017 Kaggle Competition (*Zestimate Prediction* - Zillow Prize)
- Provided dataset: 77,613 rows x 55 columns
- Target Variable: **Assessed Tax Value** (`taxvaluedollarcnt`)

Key Dataset Characteristics

- **Mix of Features**: Numerical & Categorical
- **Key Numerical Features**:
  - Bedrooms, Bathrooms, Total Living Area, Lot Size, Property Tax Values, Year Built
- **Mapped ID Features**: External dictionaries created for better interpretation (e.g., Property Land Use Type, FIPS codes)

Summary

- Dataset focuses on structural, financial, and geographic attributes.
- Well-structured dataset, ready for cleaning and preprocessing to support modeling.

Methodology: Feature Engineering

Key Transformations and New Features

- **Property Size**:
  - Log-transformed `calculatedfinishedsquarefeet`
  - Outlier flag (95th percentile)
  - Quartile indicators
- **Age Features**:
  - Property age and decade groupings
- **Ratio Features**:
  - Bedrooms/Bathrooms ratio
  - Garage presence adjusted by bedrooms
- **Binary Flags**:
  - Garage, Air Conditioning, Storage Shed

Feature Selection Decisions

- Log transformation applied to skewed features
- External dictionaries used for categorical mappings
- Simple binary flags favored for interpretability

Feature Consolidation Examples

- Fireplace-related fields combined into `fireplaceflag_new`
- Garage-related fields combined into `hasgarage_flag`
- Pool-related sparse features consolidated into `haspool_flag`

Impact Summary

- **Worked Well**:
  - Log-transformed property size, binary flags, age groupings improved model performance
- **Didn't Work**:
  - Log-transforming the target variable
  - Complex interaction terms increased overfitting

Methodology: Analytical Framework and Model Selection

Top Three Models Thus Far:

- Initial Mean CV RMSE was used to determine the three top models
  - Gradient Boosted Trees (Full features, no log)
  - Random Forest (Full features, no log)
  - Linear Regression (Final Features part 3)

Hyperparameter Tuning / Model Selection Approach:

- Top performing models underwent hyperparameter tuning.
- Key hyperparameters were chosen using scikit-learn documentation and Boston University resources.
- Each model was cross validated with tuned hyperparameters across:
  - The full feature set
  - The full feature set with a logged target
  - The best selected features from feature selection

Final Model Selection:

- Both *Training RMSE* and *Mean Cross Validation RMSE* were evaluated holistically across *all model runs* to select the model most likely to generalize and deliver best performance.

Results

Best CV RMSE: Random Forest Fine-Tuning No Log – $401,543

Best Training RMSE: Random Forest Fine-Tuning No Log – $192,416

Notable gap between Training and CV RMSE (~192K vs ~401K) indicates moderate overfitting

Random Forest Fine-Tuning No Log achieved the strongest performance on unseen data and was selected as the final model based on its superior overall performance.

Model / Run	Training RMSE	CV RMSE
GBT Fine-Tuning Log	$380,740	$416,126
GBT Fine-Tuning No Log Final	$320,343	$402,017
Linear Regression Fine-Tuning Best Features	$415,922	$415,746
Linear Regression Fine-Tuning No Log	$411,627	$412,043
Random Forest Fine-Tuning Best Features	$192,254	$407,652
Random Forest Fine-Tuning Log	$256,681	$420,191
Random Forest Fine-Tuning No Log	$192,416	$401,543

Evaluation - Random Forest

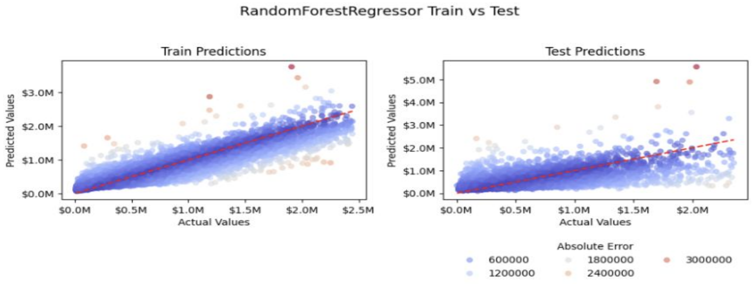
Good overall prediction, but greater dispersion for high-value properties (> $1.5M).

Key Error Patterns

- Higher absolute errors for expensive properties, especially those > $2M.
- Residual outliers remained even after IQR filtering, suggesting traditional outlier removal may be insufficient for long-tailed distributions.
- Simpler features outperformed highly complex engineered terms, reinforcing the value of model simplicity.

Future Work Recommendations

- Reduce less informative features to mitigate overfitting.
- Apply stronger outlier capping or explore robust modeling approaches.



Out[ ]:

	model	model_params	mean_CV_RMSE	std_CV_RMSE	training_RMSE	test_RMSE	run_comment
0	RandomForestRegressor	{'bootstrap': True, 'ccp_alpha': 0.0, 'criteri...	401819.224788	37547.383296	191983.475372	351428.642486	Random Forest Fine-Tuning No Log

BOSTON UNIVERSITY

Conclusion

Top Models:

Gradient Boosted Trees (GBT) and Random Forest outperformed baseline RMSE.

Random Forest outperformed GBT on the test set, although some overfitting was observed.

Key Insights:

- Models without log-transforming the target performed better.
- Random Forest demonstrated stronger generalization across property value ranges.

Practical Implications:

- Investors: More accurate tax predictions support better profitability analysis.
- Homeowners: Improved forecasts assist in financial planning and reduce uncertainty.
- Policymakers: Better evaluation of tax fairness across property types.

Next Steps:

- Fine-tune Random Forest hyperparameters to enhance robustness.
- Revisit data transformations and feature engineering to address value distribution skewness.