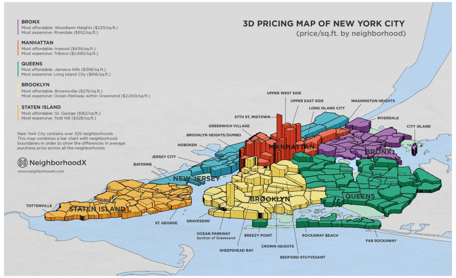
A Machine Learning Approach to the Analysis of Real Estate Prices in the New York Metropolitan Area

Group Members: Jacky He, Zhiduo Xie, Shou-Kai Cheng, Yibei Li, Yuchuan Zhang

INFO 5368: Practical Applications in Machine Learning (PAML)

May 13, 2024

1. Introduction



Motivation:

- Tackling the problem of **predicting real estate prices in NYC**.
- Important for enhancing market transparency and informed decision-making.
- Developing an application for housing price prediction using advanced ML techniques.

Technical Focus:

- Using three ML algorithms: **Linear Regression**, **Random Forest Regression**, and **Gradient Boosting Regression**.
- These models are suited for large, complex NYC real estate data.
- Unique approach combining strengths of different algorithms for better accuracy.

Impact:

- Improves access to real estate data and pricing knowledge.
- Promotes a transparent and fair market.
- Ethical concerns: ensuring data privacy and avoiding bias reinforcement in models.

2. Background

Review of Prior Work:

- Prior research used **simpler statistical models** like linear regression.
- Advanced algorithms like Ridge Regression, Random Forest, and Gradient Boosting are now used.
    - [1] Lu et al. : Hybrid approach with Lasso regression and Gradient Boosting enhanced accuracy
    - [2] Park and Bae: Evaluated multiple ML algorithms for housing price in Fairfax County

Comparison with Proposed Work:

- Our work uses Linear Regression, Random Forest, and Gradient Boosting for NYC real estate price prediction.
- Focus on direct comparison and evaluation of multiple models.
- Leverages each model's strengths to handle NYC's market complexities.

[1] Sifei Lu, Zengxiang Li, Zheng Qin, XuLei Yang, and Rick Goh. A hybrid regression technique for house prices prediction. pages 319–323, 12 2017.

[2] Bobae Park and Jae Kook Bae. Using machine learning algorithms for housing price prediction: The case of fairfax county, virginia housing data. *Expert Systems with Applications*, 42(6):2928–2934, 2015.

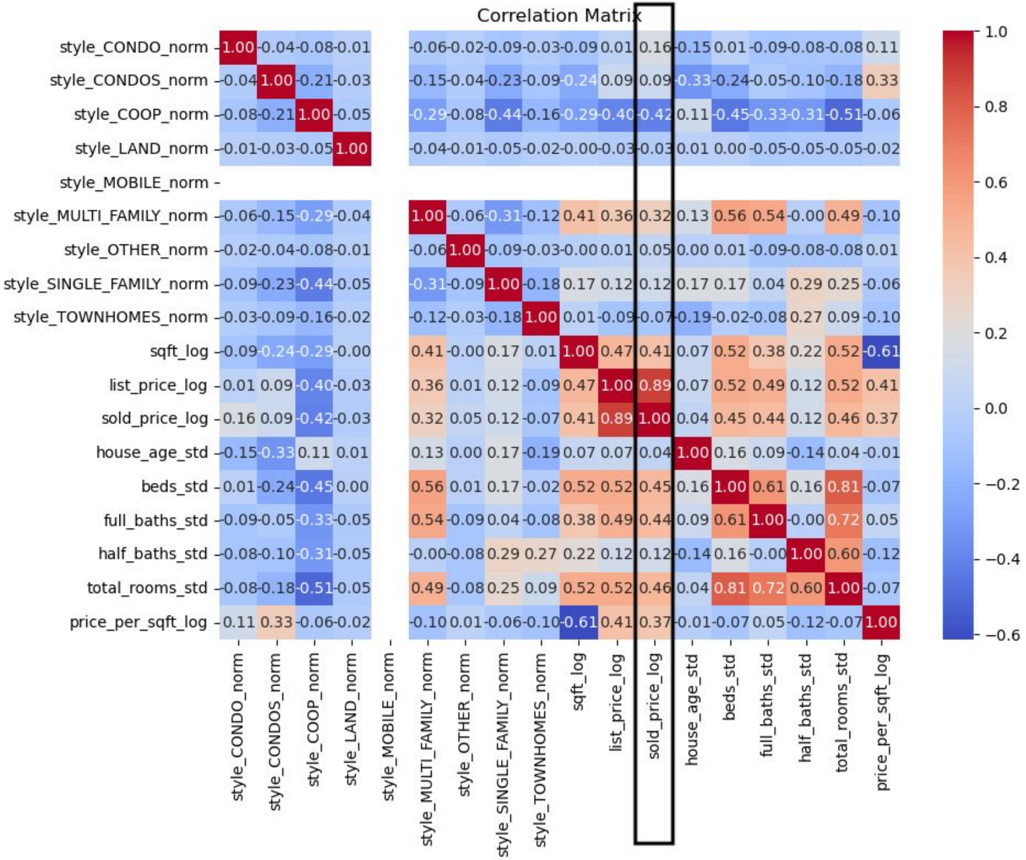
3. End-to-End ML Pipeline
3.1 Data Collection, Exploration & Processing

- **Dataset Used:** The dataset comprises property sales data from New York, NY, collected from Realtor.com via API calls. It is tailored for developing predictive models for real estate prices.
- **Data Types & Quantity:**
    - The dataset contains 10,000 records across 21 features in original. These include:
        - Categorical data: `'status'`, `'style'`, `'city'`, `'county'`
        - Numerical data: `'zip_code'`, `'beds'`, `'full_baths'`, `'half_baths'`, `'sqft'`, `'year_built'`, `'days_on_mls'`, `'list_price'`, `'sold_price'`, `'assessed_value'`, `'estimated_value'`, `'lot_sqft'`, `'price_per_sqft'`, `'latitude'`, `'longitude'`, `'stories'`.
    - This diverse set allows for comprehensive analyses of multiple aspects affecting property values.
- **Release & Source:** Data is extracted real-time covering the last 365 days from Realtor.com(May 1st, 2024), ensuring recent trends are captured for analysis.

3. End-to-End ML Pipeline
3.1 Data Collection, Exploration & Processing

May consider these factors in model training:

```
'style_MULTI_FAMILY'
'sqft'
'list_price'
'beds'
'full_baths'
'total_rooms'
'price_per_sqft'
```



3. End-to-End ML Pipeline
3.1 Data Collection, Exploration & Processing

1. **Handling Missing Data:**

- Filling missing values in the `'half_baths'` column with zeros.
- Dropping columns with significant missing values or less relevance (e.g., `'days_on_mls'`, `'assessed_value'`, `'estimated_value'`, `'lot_sqft'`, `'stories'`).
- Filling missing values in `'sqft'` with the median value of the column.
- Using the median or mode to impute missing values in other columns like `'full_baths'`, `'beds'`, `'year_built'`, `'list_price'`, `'zip_code'`, and `'county'`.
- Geographically imputing missing latitude and longitude based on median values from the same zip code.

2. **Data Cleaning:**

- Removing or imputing outliers in geographical data, ensuring that the `'latitude'` and `'longitude'` fall within specific bounds relevant to New York City.

3. **Type Conversions:**

- Converting `'zip_code'` from float to integer after filling missing values.

3. End-to-End ML Pipeline
3.1 Data Collection, Exploration & Processing

4. **Feature Engineering:**

- **Age Calculation:** A new feature `'house_age'` is created by subtracting the year_built from the current year (2024).
- **One-Hot Encoding:** The `'style'` column is transformed into multiple binary columns through one-hot encoding, ensuring that this categorical data can be used in numerical modeling.

5. **Transformation Functions Applied:**

- **Logarithmic Transformation:** Applied to `'sqft'`, `'list_price'`, and `'sold_price'`. The new features created are `'sqft_log'`, `'list_price_log'`, and `'sold_price_log'`.
- **Standardization:** Applied to `'house_age'`, `'beds'`, `'full_baths'`, and `'half_baths'`. The new features created are `'house_age_std'`, `'beds_std'`, `'full_baths_std'`, and `'half_baths_std'`.
- **Normalization:** Applied to the new `'style_'` features created by one-hot encoding. New features are named like `'style_[feature]_norm'`, where [feature] is the specific style category.

3. End-to-End ML Pipeline
3.1 Data Collection, Exploration & Processing

6. **Mathematical Feature Creation:**
   - **`'total_rooms_std'`:** Summarizes the total standardized counts of bedrooms and bathrooms (`'beds_std'`, `'full_baths_std'`, and `'half_baths_std'`) into a single feature, which provides a scaled representation of overall room count that could affect house valuation.
   - **`'price_per_sqft_log'`:** Computes the logarithm of price per square foot by dividing `'list_price_log '`by `'sqft_log'`. This feature normalizes the price per unit area, making it easier to compare properties of different sizes.

7. **Outlier Removal:**
   - Outlier removal is applied to `'sqft_log'`, `'list_price_log'`, `'sold_price_log'`, and `'price_per_sqft_log'`. Outliers are identified using the Interquartile Range (IQR) method, which defines outliers as observations that fall below Q1 - 1.5 * IQR or above Q3 + 1.5 * IQR. Removing outliers helps in reducing the impact of extreme values on the modeling process, leading to more robust and generalizable results.

CORNELL TECH
HOME OF THE JACOBS INSTITUTE

3.2 Methods and Model Training

Machine Learning Techniques:

- Implemented **Linear Regression** (from scratch) to establish a baseline for performance.
- Used **Random Forest** for its ability to handle non-linear relationships.
- Applied **Gradient Boosting Regression** to leverage sequential learning and improve accuracy.

Model Inputs and Outputs:

- Inputs: `'sqft_log'`, `'beds_std'`, `'full_baths_std'`, `'total_rooms_std'`, `'zip_code'` (based on correlation matrix)
- Outputs: `'sold_price_log'`

Training and Validation Procedure:

- Split data into 80% training and 20% testing sets to evaluate model performance.
- Used Cross-validation during training to ensure robustness and reliability.

Avoiding Overfitting and Underfitting:

- Applied regularization techniques in Linear Regression to prevent overfitting.
- Tuned Random Forest and Gradient Boosting models with parameters like tree depth and number of estimators to balance bias and variance.

3.3 Model Evaluation

Evaluation Methods and Metrics:

- Predictive accuracy: **Mean Absolute Error (MAE)** and **Root Mean Squared Error (RMSE)**.
- Explanatory power: **R-squared (R^2)** to measure the proportion of the variance in the dependent variable that is predictable from the independent variables.
- Hyperparameter settings: e.g. learning rate, iterations, max depth, number of trees etc.

```
class LinearRegressionFromScratch:
    def __init__(self, learning_rate=0.01, iterations=1000):
        self.learning_rate = learning_rate
        self.iterations = iterations
```

```
param_grid_rf = {
    'n_estimators': [100, 200, 300],
    'max_depth': [None, 10, 20, 30],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4],
    'max_features': ['auto', 'sqrt', 'log2']
}
```

```
param_grid_gb = {
    'n_estimators': [100, 200, 300],
    'learning_rate': [0.01, 0.1, 0.2],
    'max_depth': [3, 5, 7],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4],
    'subsample': [0.8, 1.0]
}
```

	Linear Regression 👎	Random Forest 👍	Gradient Boosting
RMSE	0.45	0.31	0.32
MAE	0.35	0.22	0.24
R^2	0.33	0.68	0.64

3.4 Model Deployment

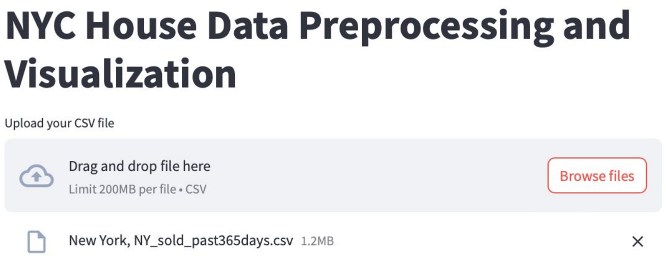
Application of the proposed project including user inputs and model outputs.

- **Price Estimation:** Uses advanced algorithms (Linear Regression, Random Forest, Gradient Boosting) to predict real estate prices.
- **Market Transparency:** Improves transparency and supports informed decisions in NYC's real estate market.
- **Market Stability:** Aids both buyers and sellers with accurate forecasts, stabilizing the market and widening access to real estate investments.

Application Importance and Ethical/Societal Implications

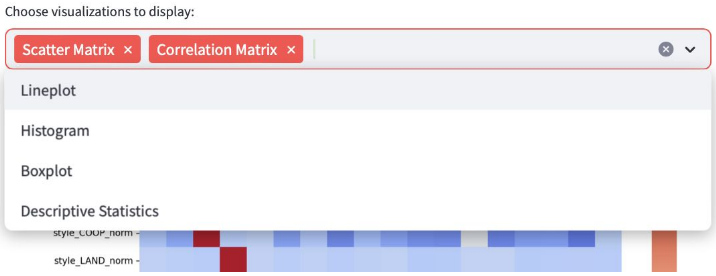
- **Democratizes Access:** Improves market transparency and equity by making complicated real estate data available.
- **Data Privacy:** Secures user data to protect personal information.
- **Bias Mitigation:** Updates and refines models to prevent perpetuating market inequities.
- **Techniques for Fairness:** Employs cross-validation and regularization to enhance model reliability and compliance with data protection laws.

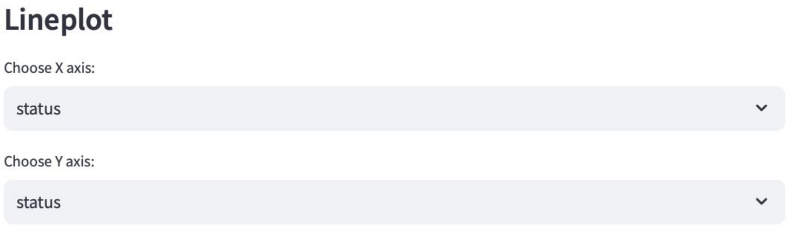
3.5 Front-End (Streamlit)



Dataset upload

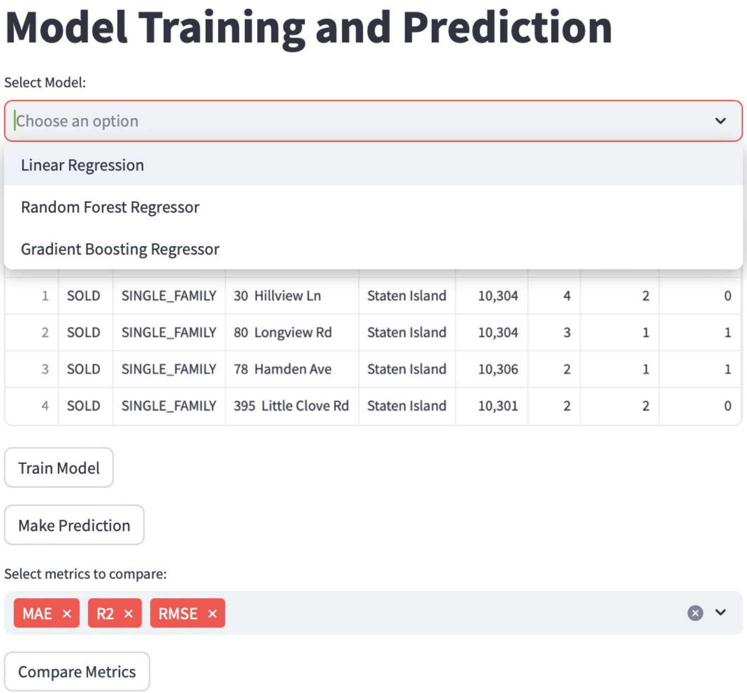
Choose visualizations to display





Choose X and Y axis

3.5 Front-End (Streamlit)



- Choose one model from select box
- Train Model button
- Make Prediction button
- Metrics select box
- Compare Metrics button

4. Results

Expected outcome based on prior work or a high-level inspiration or motivation:

- **Visualization Dashboard:** Features a user-friendly interface that displays insights from advanced machine learning models like Linear Regression, Random Forest, and Gradient Boosting.

- **Data Handling:** Utilizes an ETL process to manage complex NYC real estate datasets, enabling precise housing price predictions.

- **Market Insights:** Delivers in-depth market dynamics and simplifies complex data for user convenience.

- **Future Enhancements:** Plans to expand data sources and introduce more predictive models for increased accuracy and functionality.

CORNELL TECH
HOME OF THE JACOBS INSTITUTE

Thank you :)