**Module-03**
**Individual task**

# Feature Extraction Thought Experiment: Select a dataset (e.g., photos, shopping lists) and describe which features would be important to a machine learning model.

## 1. Executive Summary

This report details a thought experiment regarding feature extraction for a machine learning (ML) model designed to predict residential housing prices. Accurate price prediction is a regression problem heavily dependent on the quality and relevance of input features. While raw data provides the foundation, the value lies in extracting meaningful signals from structured, unstructured, and geospatial sources. This document outlines the selected dataset, identifies critical feature categories, proposes extraction methodologies, and addresses ethical considerations regarding bias. The objective is to demonstrate how thoughtful feature extraction directly influences model performance and fairness.

## 2. Introduction and Problem Statement

### 2.1 The Objective

The goal is to build a supervised learning model to estimate the market value of single-family homes. The target variable is the final sale price.

### 2.2 The Dataset

For this experiment, we select a composite dataset similar to those found on platforms like Zillow or Redfin. The raw data includes:

Structured Tabular Data: Square footage, number of bedrooms/bathrooms, year built, lot size.

Unstructured Text: Property descriptions, agent remarks, school district names.

Image Data: Exterior and interior photographs.

Geospatial Data: Latitude/longitude coordinates, neighborhood boundaries.

Temporal Data: Listing date, days on market, historical tax assessments.

### 2.3 The Challenge

Raw data is rarely model-ready. A column labeled "description" contains noise and signal mixed together. A timestamp needs conversion to "age of property." Coordinates need conversion to "distance to city center." This report focuses on the extraction phase: transforming raw inputs into informative features.

## 3. Feature Extraction Strategy

To maximize predictive power, we categorize features into four distinct domains. Each requires a specific extraction technique.

### 3.1 Structural and Physical Attributes

Source: Structured Tabular Data

While some data is ready-to-use, derived features often hold more predictive weight than raw counts.

Raw Input: `Total_SqFt`, `Num_Bedrooms`, `Num_Bathrooms`, `Year_Built`.

Extraction Logic:

Price Per Square Foot: While this uses the target variable in training, for inference, we extract `SqFt` as a normalization factor for comparison.

Property Age: Extract `Current_Year - Year_Built`. A 1920s home has different value drivers than a 2020s home.

Room Ratio: Extract `Num_Bedrooms / Total_SqFt`. High bedroom counts in low square footage indicate smaller rooms, potentially lowering value.

Renovation Indicator: If `Year_Built` differs significantly from `Last_Renovation_Year`, extract a binary flag `Is_Renovated`.

### 3.2 Geospatial and Location Context

Source: Latitude/Longitude and Map APIs

Location is the primary driver of real estate value. Raw coordinates are useless to a model without context.

Raw Input: `Latitude`, `Longitude`.

Extraction Logic:

Distance to CBD (Central Business District): Calculate the Haversine distance from the property to the city center.

Amenity Proximity: Extract distances to the nearest top-rated school, grocery store, and public transit station.

Neighborhood Cluster: Use clustering algorithms (e.g., DBSCAN) on coordinates to group properties into micro-neighborhoods, extracting a `Cluster_ID` as a categorical feature.

Flood/Zone Risk: Cross-reference coordinates with FEMA flood maps to extract a binary `Flood_Zone` flag.

### 3.3 Semantic and Sentiment Analysis

Source: Unstructured Text Descriptions

Agent descriptions contain subjective quality indicators not captured in square footage.

Raw Input: "Stunning renovated kitchen with granite countertops and hardwood floors throughout."

Extraction Logic:

Keyword Frequency (Bag of Words/TF-IDF): Extract presence of high-value keywords: "granite," "hardwood," "stainless steel," "view," "pool."

Sentiment Score: Use NLP models (e.g., BERT or VADER) to extract a sentiment polarity score. Overly hyperbolic language might indicate a property that has been on the market too long.

School District Quality: Extract the numerical rating from text strings (e.g., "Serving Lincoln High (9/10)") to create a `School_Rating` integer.

3.4 Visual Quality Assessment

Source: Property Images

Photos reveal condition, which is a massive value driver.

Raw Input: 20 JPEGs per listing.

Extraction Logic:

Room Classification: Use a Convolutional Neural Network (CNN) to tag images (Kitchen, Bathroom, Living Room, Exterior).

Quality Embeddings: Pass images through a pre-trained network (e.g., ResNet) and extract the feature vector from the penultimate layer. This vector represents "visual style" and "condition" numerically.

Brightness and Clutter: Extract low-level features like average pixel brightness (dark photos may imply poor maintenance) or edge detection density (clutter).

Curb Appeal Score: Aggregate exterior image embeddings into a single `Curb_Appeal` score (0-1).

## 4. Feature Selection and Dimensionality Reduction

Extraction often leads to high dimensionality (e.g., hundreds of keyword flags or image vectors). Selection is required to prevent overfitting.

4.1 Correlation Analysis

We will compute a Pearson correlation matrix. If `Num_Garage_Cars` and `Garage_SqFt` have a correlation > 0.9, we drop one to reduce multicollinearity.

4.2 Importance Ranking

Using a tree-based model (e.g., Random Forest) on a subset of data, we will rank features by Information Gain. Features contributing less than 1% to the reduction in impurity will be candidates for removal.

4.3 Principal Component Analysis (PCA)

For the image embeddings (which may be 512+ dimensions), we will apply PCA to reduce them to 10-20 principal components that capture 95% of the variance, preserving visual information while reducing computational load.

## 5. Ethical Considerations and Bias Mitigation

In housing prediction, feature extraction carries significant ethical risk. Certain features can act as proxies for protected classes, leading to algorithmic redlining.

5.1 The Zip Code Problem

Risk: Using `Zip_Code` or `Neighborhood_Name` can inadvertently encode racial or socioeconomic segregation history.

Mitigation: Instead of using neighborhood names, rely on objective geospatial features (distance to transit, school rating). If neighborhood data is essential, audit the model to ensure error rates are consistent across different demographic areas.

5.2 Image Bias

Risk: Visual models might devalue homes based on decor styles associated with specific cultures or socioeconomic statuses.

Mitigation: Focus image extraction on structural condition (cracks, water damage) rather than style (furniture, art).

5.3 Textual Bias

Risk: Descriptions might use coded language (e.g., "safe neighborhood" implying lack of diversity).

Mitigation: Filter out subjective safety language during the NLP extraction phase. Focus on physical amenities only.

## 6. Implementation Roadmap

To move from thought experiment to production, the following steps are recommended:

1. Pipeline Construction: Build an ETL pipeline that automates the extraction logic defined in Section 3.
2. Baseline Modeling: Train a model using only structured data (SqFt, Beds, Baths) to establish a baseline RMSE (Root Mean Square Error).
3. Iterative Addition: Incrementally add Geospatial, Text, and Image features. Measure the delta in RMSE for each addition to justify the computational cost.
4. Bias Audit: Run disparate impact analysis on the final model before deployment.

## 7. Conclusion

Feature extraction is the bridge between raw data and machine intelligence. In the context of real estate pricing, relying solely on structured data (bedrooms and bathrooms) yields a rudimentary model. By extracting signals from text (amenities), images (condition), and geography (location context), we significantly enhance predictive accuracy.

However, this power comes with responsibility. As demonstrated in Section 5, features that improve accuracy can also encode historical biases. A successful ML strategy must balance performance with fairness. By prioritizing objective physical and geospatial features over subjective or demographic proxies, we can build a model that is both accurate and equitable.

This thought experiment confirms that the "secret sauce" of machine learning is not merely the algorithm chosen, but the ingenuity applied to extracting and engineering the input features.