Dynamic Investment Models in Real Estate: A Machine Learning Approach

Master Thesis

Author: Yassine Hammou (Student ID: 7354517)
Supervisor: Prof. Dr. Markus Weinmann
Co-Supervisor: Sercan Demir

Faculty of Management, Economics and Social Sciences
University of Cologne

October 20, 2024

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne die Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten und nicht veröffentlichten Schriften entnommen wurden, sind als solche kenntlich gemacht. Die Arbeit ist in gleicher oder ähnlicher Form oder auszugsweise im Rahmen einer anderen Prüfung noch nicht vorgelegt worden. Ich versichere, dass die eingereichte elektronische Fassung der eingereichten Druckfassung vollständig entspricht.

Die Strafbarkeit einer falschen eidesstattlichen Versicherung ist mir bekannt, namentlich die Strafandrohung gemäß § 156 StGB bis zu drei Jahren Freiheitsstrafe oder Geldstrafe bei vorsätzlicher Begehung der Tat bzw. gemäß § 161 Abs. 1 StGB bis zu einem Jahr Freiheitsstrafe oder Geldstrafe bei fahrlässiger Begehung.

Yassine Hammou

Köln, den xx.xx.20xx

Abstract

This study introduces a novel business model that leverages machine learning techniques to offer an alternative financing option for homeowners by selling the shares in property appreciation. Profitability is evaluated through various machine learning algorithms, including gradient boosting and k-nearest neighbors, along with feature extraction methods such as principal component analysis (PCA) and clustering. To enhance interpretability, feature importance metrics like SHAP values and permutation importance are utilized to assess the significance of different features. The study reveals that key predictors of house price appreciation include the features living area, the county's average household income, and geographic factors. Regarding the profitability of the business model, results indicate that an annual investment growth rate of 27.86% is possible for investors. However, the model exhibits a tendency to underpredict initial house prices, leading to borrowers overpaying investors by an average of 9%. These findings suggest adjustments are needed to enhance fairness. Despite this, the proposed model presents a promising alternative to traditional financial systems, with potential benefits for both investors and homeowners.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	**Full Form**
AI	Artificial Intelligence
AdaBoost	Adaptive Boosting
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
CRISP-DM	Cross Industry Standard Process for Data Mining
GBDT	Gradient Boosting Decision Trees
GIS	Geographic Information System
HPI	House Price Index
KFF	Kaiser Family Foundation
KNN	K-nearest Neighbors
MAPE	Mean Absolute Percentage Error
MSE	Mean Squared Error
NLP	Natural Language Processing
PCA	Principal Component Analysis
PC	Principal Component
RFE	Recursive Feature Elimination
RMSE	Root Mean Squared Error
SHAP	Shapely Additive Explanations
SMAPE	Symmetric Mean Absolute Percentage Error
SVD	Singular Value Decomposition
SVM	Support Vector Machine
USA	United States of America
XGB	Extreme Gradient Boosting

1 Introduction

1.1 Motivation and Contribution to Research

The United States of America struggles with a profoundly flawed social system, where access to vital necessities like health insurance is often tied to employment and is overall very costly. According to the Kaiser Family Foundation (KFF), the most common subtype of health insurance coverage, with 54.5% of the American population, was employment-based health insurance. The top reason for non-elderly adults (ages between 18 and 64) being uninsured is the high health insurance costs, according to the statistics from the Centers for Disease Control and Prevention ("Reasons for Being Uninsured Among Adults", 2020). About 41% of adults in the USA besides that have debt from unpaid medical or dental bills ("Debt - KFF Health News", n.d.).

This underscores the reality that being insured is a luxury for many in the USA, with high treatment and medical costs pushing individuals to the brink of financial ruin. As a result, losing a job jeopardizes one's livelihood and strips away essential benefits, including health insurance, leaving individuals vulnerable to unforeseen crises. Compounded by the burden of mortgage debt, with the average debt an American owes being $104,215 and mortgage loans making up the most significant fraction, the risk of homelessness looms large (Streaks, 2024).

Losing a job can, therefore, be detrimental for homeowners with mortgages and debts. Without any savings, this can lead to a financial crisis, with very limited solutions to find quick financial support to bridge the time until finding another job or another source of income.

One option is to obtain loans, which, while common, can lead individuals further into debt, worsening their financial situation. Apart from that, it is almost impossible to get another loan for individuals already in debt and without current income sources. Borrowing from friends, though well-intentioned, is not always feasible and would most probably not cover all expenses.

The solution of selling one's home, a last resort, comes with its own challenges, from prolonged waiting periods to the uncertainty of finding adequate housing thereafter. According to the United States Interagency Council on Homelessness, affordable housing is scarce. Only 37 affordable homes are available for every 100 low-income renters. Low-income households have to spend more than half their income on rent ("Homelessness Data & Trends", n.d.).

Seeing this, there is clearly a need for a novel approach to bridge this gap of financial insecurity. The new approach would require providing quick access to financial means without causing an immediate debt spiral. However, to be

self-sustaining and without being dependent on people's kind-heartedness, the approach also has to be profitable for the party that provides the financial means.

This thesis introduces and, at the same time, tests a new business model that aims to close this gap of missing financing options. The key idea of the business model is to sell shares in property appreciation and, therefore, provide an alternative financing option for homeowners. In this thesis, we test this business model's profitability and try to back up the investment decision-making process using machine learning models. Testing the profitability of the business model is crucial to get insights into whether it is an interesting concept for potential investors. Overall, we therefore want to answer the following research question:

Can the novel machine learning-based business model serve as a viable alternative to the traditional financial system by offering both an attractive investment opportunity for investors and a less debt-intensive financing option for homeowners?

Through empirical analysis and data-driven insights, this research seeks to not only address this question but also pave the way for a more equitable and resilient future, offering homeowners a pathway to financial security without the burden of exorbitant debts. This research not only has practical applications but also contributes to research.

This research extends the frontier of financial innovation by testing a novel business model that leverages property appreciation shares and machine learning. It offers fresh insights into alternative financing mechanisms that can potentially mitigate financial vulnerabilities. Using a data-driven and machine learning-based approach, this research provides valuable evidence regarding the viability and effectiveness of such innovative approaches. Through bridging the domains of finance, machine learning, and housing economics, the value of interdisciplinary collaboration in addressing complex problems is demonstrated, underscoring the importance of integrating diverse perspectives and expertise to develop holistic solutions.

It also inspires further research and innovation in the mentioned fields. Researchers and practitioners may build upon the findings presented in this thesis to explore new avenues for enhancing the proposed business model.

1.2 Structure of the Paper

This thesis is divided into nine chapters, each with corresponding subchapters. The first chapter serves as an introduction, outlining the motivation behind the research project. This section defines both the practical and research problems, leading to the formulation of the central research question: whether a novel busi-

ness model leveraging machine learning can offer a viable alternative to traditional financial systems. Additionally, the potential contributions to academic research and practical application are outlined.

The second chapter introduces and describes the novel business model examined in this thesis.

The third chapter, dedicated to the theoretical background, establishes the conceptual and terminological foundations necessary for this study. This includes defining machine learning and exploring various machine learning architectures and concepts.

The fourth chapter reviews the current state of research in related fields, focusing on house price prediction and house price appreciation prediction.

The fifth chapter discusses the methodology, including the underlying methodological framework and the chosen research approach.

In the sixth chapter, the data sources utilized are described, along with the steps taken for data preparation prior to analysis and modeling.

The seventh chapter systematically presents the results, including an in-depth data analysis, a description of the architecture and performance of different machine learning models, and an evaluation of the proposed business model.

The eighth chapter, the discussion, critically examines the results presented in the previous chapters, reflecting on their contributions to research and practice. It also addresses the limitations of the research, particularly regarding the methodological approach, and suggests new questions that could guide future research.

Finally, the ninth chapter concludes the thesis, evaluating the achievement of the research objectives in light of the presented results. This chapter provides a reflective summary of the study, highlighting key findings and their implications.

2 Business Model

2.1 Framework

The fundamental idea of the business model revolves around acquiring shares in the appreciation of properties.

Consider a scenario where a homeowner requires financial assistance and seeks to raise capital. Conventionally, options would include securing a loan from a bank or selling the property. However, this business model presents an alternative financing option.

At its core, this model includes two basic concepts: lending money and profit sharing, combining them into a novel framework. Here's how it works:

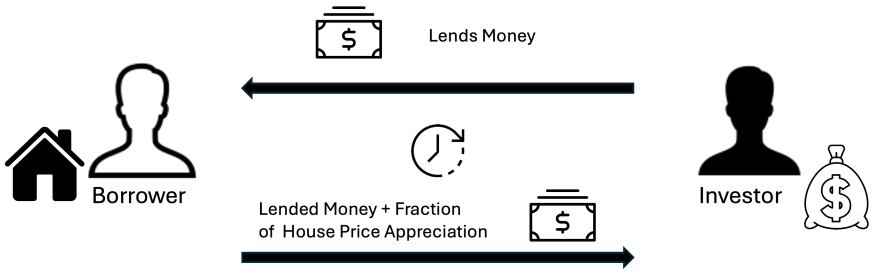


Figure 1: Business Model Framework

Two parties are involved: the property owner and the investor. The property owner seeks financial assistance without incurring debt, while the investor aims to identify a secure yet lucrative investment opportunity.

The investor gives a specified sum of money to the property owner, who is obligated to repay this amount within a predetermined timeframe. Thus far, this resembles a conventional lending scheme. However, the innovative aspect of this business model lies in the additional provision: the property owner must also pay the investor a portion of the property's appreciation during the loan tenure.

In essence, if the property's value remains constant, the property owner incurs no additional obligation beyond repaying the borrowed sum. This unique feature ensures that the investor's investment remains secure as they are guaranteed to get back their initial capital. Even in a worst-case scenario, such as a decline in the property's value, the investor avoids losses, unlike the risks associated with direct property ownership.

For the homeowner, this model offers a compelling alternative to property sale for accessing funds. The supplementary payment owed to the investor represents a fraction of the realized profit from the property's increased value during the loan period. Consequently, the homeowner only shares a portion of the accrued profit, which would have otherwise been foregone if the property were sold. Thus, the business model embodies the principle of profit-sharing, aligning incentives between the homeowner and the investor.

In summary, this business model innovatively blends lending principles with property profit sharing, offering a mutually beneficial solution that mitigates financial risks for both parties involved.

2.2 Integration with Machine Learning

To ensure the business model operates effectively, it is essential to accurately estimate the property's market price or potential sales price, reflecting the current market conditions. This accuracy is crucial for correctly determining the property's true value appreciation, thereby ensuring that the profit from this

appreciation can be realized through market sales.

If we significantly undervalue or underestimate the property's worth after the lending period, it would negatively impact and reduce the calculated value appreciation, resulting in a lower return and profit for the investor. Conversely, overestimating the property's price at the end of the lending period would lead to an inflated calculated value appreciation and a property price that cannot be achieved through a market sale. This situation would force the property owner to pay the investor more than necessary. Similarly, incorrect property value estimation at the beginning of the lending period poses the same risks. Therefore, it is vital for the business model's success that both parties can trust the accurate property value estimation.

To achieve this, we must first develop a machine learning model that accurately estimates the market value of a property (house price), taking into account current economic factors and conditions. This model must be capable of dynamically adjusting the property value to ensure precise estimations at all times.

When faced with multiple property options, investors logically choose the property expected to increase the most in value to maximize profit. Thus, it is necessary to build a machine learning model that can estimate a property's potential future value appreciation. Like the house price prediction model, this model should also account for the current economic situation.

In this context, the precise estimation of true value appreciation is less critical to the business model's functionality than it is for the house price prediction model. Instead, it is sufficient to indicate the best property to invest in from a set of options through an appropriate estimation of value appreciation. This model primarily benefits the investor and is less important to the property owner.

3 Method

Our study uses the Cross Industry Standard Process for Data Mining (CRISP-DM). It was introduced by Wirth and Hipp (2000) and provides a structured approach for data mining projects.

To leverage the CRISP-DM methodology, we translate our business problem of making investment decisions into a data mining task. Specifically, we will use the CRISP-DM approach to develop robust predictive models for our house price prediction and price appreciation prediction problems, which we derive from our novel business model.

The framework is described as a hierarchical process consisting of different abstraction levels (Wirth & Hipp, 2000). The phase level organizes the overall process into a small number of phases. These phases consist of generic tasks, which are broken down into more specialized tasks that define the specific actions to be taken (Wirth & Hipp, 2000).

In our case, we have the generic task of building a machine learning model to predict house prices and price appreciations using historical transaction data. We take this generic task and break it down into tasks such as splitting the data into a test, validation, and train set and performing grid search to find optimal hyperparameters for a better predictive performance of our model.

The hierarchical level is the process instance level, which is a record of actions, decisions, and results. It is used to track what happened during the execution of the overall process and to document the results of the actions taken (Wirth & Hipp, 2000). In CRISP-DM, the sequence of tasks is not strictly defined, nor is it sequential. Rather, we have the flexibility to go back to earlier tasks and repeat certain actions based on the knowledge gained throughout the process.

The CRISP-DM consists of the following six phases:

1. Business Understanding
2. Data Understanding
3. Data Preparation
4. Modelling
5. Evaluation
6. Deployment

In our research project, we iteratively go through these. Therefore, we can successively build up our domain knowledge in the real estate valuation field and better understand the data and connected challenges such as data quality. We aim to build our initial model early on using a simple model architecture to use

the results and performance of this model as a benchmark to understand the baseline performance and gain insights. Based on these insights, we generate ideas to improve the predictive performance and develop more sophisticated model architectures. We iterate through the process, making changes and improvements in subsequent iterations. By closely monitoring performance changes and evaluating their impact, we isolate and analyze the causes of these changes. This iterative approach allows us to continually refine our models and strive for the best possible solution for the reliable prediction of house prices and price appreciations.

The CRISP-DM methodology provides a valuable framework for planning, documenting, and communicating throughout the research project (Wirth & Hipp, 2000). It ensures that we follow a structured and systematic approach, allowing us to make informed decisions and keep track of introduced changes and their effects to address the business problem most efficiently.

4 Data

4.1 Data Sources

A substantial amount of data is generally required to build reliable machine learning models. The more data available, the better the potential performance of the models. Our objective is to develop two machine learning models: one to predict a house's sales price or market value and another to predict the potential house price appreciation over the years. Essential to this task is transactional data on house sales, which includes information about the properties, sales prices, and transaction dates. Additionally, data that records multiple sales of the same property is crucial for calculating house price appreciation over time, serving as the target for our second model.

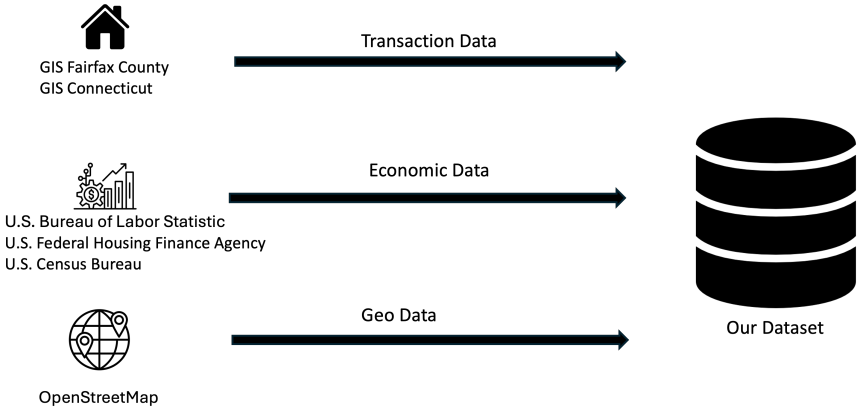


Figure 2: Data Sources

We obtained this data from publicly available databases (see Figure 2). The primary sources are the Geographic Information System (GIS) Database of Fairfax County, Virginia, and the GIS Database of Connecticut, the southernmost state in the New England region of the Northeastern United States (see Figure 3). These databases include metadata about sales transactions and information about the properties.

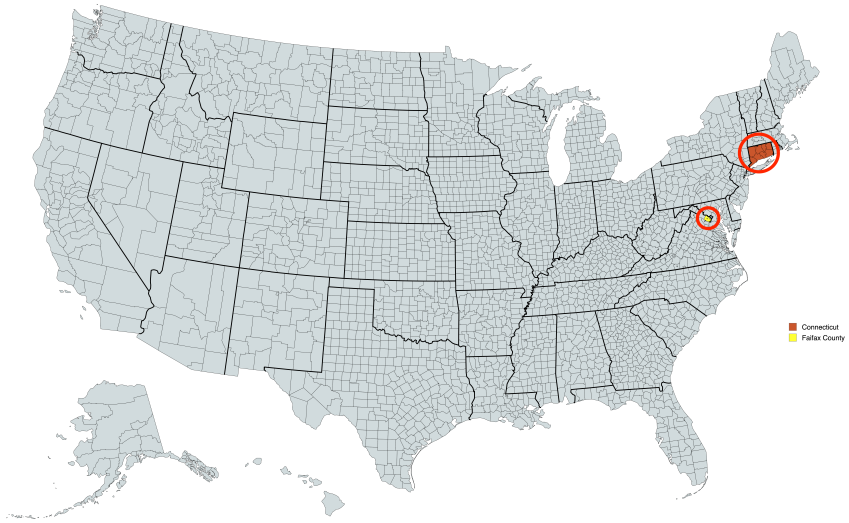


Figure 3: Location of Fairfax County and Connecticut

Economic factors, which fluctuate over the years, could also affect property prices. Thus, it is essential to consider the economic conditions at the time of each sale to enable our models to adjust property values based on the economic situation dynamically. For this, we utilized publicly available data from the U.S. Bureau of Labor Statistics, the U.S. Federal Housing Finance Agency, and the U.S. Census Bureau. The economic data gathered includes information on employment, poverty, income, the house price index, and new private housing structures authorized by building permits in the respective counties.

County	**State**
Fairfax	Virginia
Fairfield	Connecticut
Hartford	
Litchfield	
Middlesex	
New Haven	
New London	
Tolland	
Windham	

Table 2: County and State Information

Given that economic conditions vary significantly by location, we collected data specific to each county rather than relying on national averages. The counties covered include Fairfax County and the eight counties in Connecticut: Fairfield, Hartford, Litchfield, Middlesex, New Haven, New London, Tolland, and Windham (see Figure 4).

Figure 4: Counties in Connecticut

Since the transaction data from the GIS databases lacks detailed locational information (such as geocoordinates), we used address data provided in the GIS transaction data to obtain geo-information. We used OpenStreetMap, a free, open geographic database maintained by a community of volunteers through open collaboration. To calculate richer locational features, such as the distance to the nearest airport, we gathered data on amenities and their geocoordinates using OpenStreetMap's Overpass Turbo API.

4.2 Data Preparation

As described in the data sources section, we have three primary data sources: transaction data from two counties, geodata from OpenStreetMap, and economic data (see Figure 2). To prepare the data for our analysis and modeling steps, we must process and join all the data together.

The first step of data preparation involves cleaning the transaction data. Since the data came from various sources, we needed to consolidate it. We began by identifying common attributes across the datasets. Next, we standardized categorical data, such as the property's overall condition, into the categories: Poor, Average, Average Plus, Fair, and Good.

The consolidated transaction data included the following attributes: address, city, county, condition, year built, effective year built, number of bedrooms, number of bathrooms, number of half baths, living area, effective living area, and sale date. We addressed outliers in the sales price by setting upper and lower bounds at two standard deviations from the mean. Duplicates and transactions with

unrealistic prices (e.g., 0 or 1) or missing values were removed. Additionally, we calculated two new features: the age of the house at the time of sale and the effective age of the house at the time of sale, which reflects the age of the house after major renovations. These features were determined by subtracting the year of construction or renovation from the year of sale.

The next step was geocoding the property locations to obtain their geo-coordinates (latitude and longitude). We downloaded geo-coordinates for all addresses from OpenStreetMap for Fairfax County and Connecticut. The objective was to match our transaction address data with OpenStreetMap address data via a left join to append geo-coordinates. To achieve a high matching rate when joining the data, it was necessary to clean the address data, which involved removing unnecessary parts from the address, expanding abbreviations, and trimming whitespace. We also extracted street names and house numbers as separate attributes. Initially, we joined the data at the house number level (house number, street, city, county, state). We performed a secondary join at the street level (street, city, county, state) for unmatched addresses. Transactions without geo-coordinates after this process were discarded. From 805,115 unique addresses, we matched 547,303 (67.98%) on the house number level and 237,064 (29.44%) on the street level. A total of 20,748 (2.58%) addresses could not be matched.

We then engineered various geographical features, as the property location could impact the sales price. We introduced two types of features: distance measures (e.g., distance to the nearest airport) and frequency measures (e.g., the number of restaurants within a 5 km radius). Using OpenStreetMap data, we categorized amenities into groups such as religious institutions, educational facilities, healthcare facilities, emergency services, animal care facilities, community venues, shopping locations, food and drink establishments, financial services, transportation services, entertainment venues, adult entertainment, sports facilities, accommodations, government and civic services, and recreational facilities (see Table 3).

For each property and amenity category, we calculated the number of amenities within a 5 km radius to obtain the frequency measures. We also computed the distance from each property to the nearest airport, ferry terminal, railway station, marketplace, hospital, hotel, and museum to obtain the distance measures. These features were then merged with the transaction data.

Next, we incorporated economic data. The economic data includes the number of unemployed people, the unemployment rate, the number of employed people, the house price index, household income, the number of new housing permits, population, the number of people living in poverty, the poverty rate, and the number of young people living in poverty. The economic data includes key figures

Amenity Group	**Amenity**
Religious Institutions	Place of worship, monastery
Educational Facilities	School, university, college, library, kindergarten music school, prep school, driving school, childcare
Healthcare	Dentist, clinic, hospital, doctors, nursing home
Emergency Facilities	Fire station, police
Animal Care Facilities	Animal shelter, animal boarding
Community Venues	Social facility, community centre, exhibition centre conference centre, social centre, town hall, coworking space
Community Services	Charity, graveyard, crematorium, mortuary, ranger station, post depot, mail room, public bath, public bookcase
Shopping	Marketplace, market
Food and Drink	Restaurant, fast food, ice cream, café, BBQ, canteen
Financial	Bank, ATM, finance, money transfer, check cashing
Transport	Aerodrome, railway platform, ferry terminal, bus station, boat storage, bus platform, taxi, railway halt, railway car shuttle, car sharing, railway station, bus stop position, ferry stop position
Entertainment	Museum, art centre, theme park, stadium seating cinema, theatre, attraction, amusement, events venue planetarium, lounge, internet café, karaoke box, music venue
Adult Entertainment	Hookah, hookah lounge, biergarten, casino nightclub, pub, bar, love hotel
Sports	Dancing school, dojo, ski school, ski rental
Utilities	Charging station, compressed air, sanitary dump station, vacuum cleaner, waste transfer station, waste disposal
Accommodation	Motel, hotel, hostel
Governmental Civic Services	Courthouse, prison
Recreational	Park, campground, campsite, picnic site, zoo, aquarium, viewpoint, boat rental, bicycle rental

Table 3: Amenities and their groups

from 1990 to 2024.

Some of these features are yearly, and some are monthly. To ensure no data leakage, we joined the data from the month or year preceding the sale date. For monthly data, we joined on the previous month, and for yearly data, on the previous year. Since we gathered the economic data at the county level, we accordingly joined the data at the county level.

With these steps completed, the data was ready for the house price prediction model. For the appreciation prediction model, where the target variable is the appreciation of the sale price compared to the previous sale, we calculated the target value. This involved grouping the data by property and computing the percentage increase in sale prices for each subsequent sale, excluding the first sale. Additionally, we introduced a new feature: the appreciation time, which is the duration between the initial investment and the end of the investment period. In the context of our transaction data, this is the time between house sales.

The dataset for the house price prediction model, which uses the sales price as the target variable, contains 1,036,383 transactions. The dataset for the appreciation

prediction model, which focuses on value appreciation as the target variable, includes 468,439 transactions. Both datasets encompass transactions from 1900 to 2023.

5 Theoretical Background

5.1 Artificial Intelligence

The complexity of intelligence, with no established definition, results in the absence of a universally accepted definition of artificial intelligence (AI) (Monett & Lewis, 2018). In his 1950 paper *Computing Machinery and Intelligence*, Alan Turing introduced the Turing test, which aims to define when a machine can be classified as intelligent. According to this test, the machine is considered intelligent if a human interrogator cannot distinguish between a computer and a human through text interaction (Dobrev, 2012; Turing, 1950).
Russell and Norvig (2021) outline that a system needs natural language processing, knowledge representation, automated reasoning, and machine learning to pass the Turing Test. For the total Turing test, which includes perceptual abilities, it also requires computer vision and robotics.
More recent definition approaches focus on specific aspects of AI, such as the ability to mimic human skills and capabilities (Brynjolfsson & Mitchell, 2017) and the ability to learn (Castelvecchi, 2016).
This study uses the definition by Berente et al. (2021). Berente et al. (2021) define AI as "the frontier of computational advancements that reference human intelligence in addressing ever more complex decision-making problems".
In our case, we want to leverage these computational advancements in the field of AI to also tackle a complex decision-making problem in the form of investment decisions and incorporate them into a novel business model.

5.2 Machine Learning

Machine learning is a component or subfield of AI. It focuses on algorithms that enable computers to learn from data and make predictions or decisions based on that data (Jordan & Mitchell, 2015).
The key aspect is that these data-driven algorithms and models allow computers to perform tasks without explicit programming for each specific task. Machine learning models learn to perform various tasks or solve complex decision problems by identifying patterns within large datasets (Carleo et al., 2019).
Machine learning algorithms utilize various statistical and computational techniques to enhance performance for different tasks and learn from experiences, enabling the analysis of complex datasets (Mohri et al., 2012).
Machine learning encompasses various types of learning, including supervised, unsupervised, and reinforcement learning. Supervised learning algorithms learn

with guidance, meaning the true outcome tried to predict is already known during the learning process. In contrast, unsupervised learning algorithms learn without guidance, as the true outcome is not provided. Reinforcement learning is distinct from the other two types. Reinforcement learning employs a reward-based approach to maximize the overall reward to achieve the best possible result (Gupta et al., 2022). In this thesis, we concentrate on supervised learning techniques, which will be explained in more depth in the next sections.

5.3 Supervised Learning

Supervised learning operates under supervision and utilizes labeled datasets, where the outcome variables we aim to predict are already provided (in the training data) (Gupta et al., 2022). Supervised learning aims to construct prediction models for these specific outcome variables based on a set of predictor variables, also known as features. The relationship between the outcome and the features is approximated using training data, where the predictors and the outcome are known (Kern et al., 2019).
In our case, we want to predict house prices and price appreciations. To do this, we use predictor variables or features such as the age of the house, square footage, and the number of bedrooms.
The derived model can then predict the outcome for new, previously unseen observations (test data) where the outcome variable is unknown. To achieve an accurate approximation of the true function between the outcome variable and the features and to ensure prediction values are as close to the true values as possible while being resilient to variations in the data, it is necessary to optimize the model configuration (model tuning) and select the best model from different approaches based on performance on unseen data.
Validation techniques such as cross-validation are used to better estimate the model's performance on real-world data. Cross-validation provides out-of-sample prediction performance on data not used during the training process and, therefore, not seen by the model before (Kern et al., 2019).
Overall, there are two different categories of supervised learning, which differ based on the possible values of the outcome variable. One category is classification, where the outcome variable is binary or multi-class, meaning it can take only a limited set of values. The other category is regression, where the outcome variable is continuous and not limited to a specific finite set of values (Gupta et al., 2022).
Since we aim to predict house prices and value appreciations, which are continuous values, we will focus on regression algorithms in this paper.

While various supervised learning methods can be applied in predictive settings, tree-based approaches may be especially advantageous.

5.4 Supervised Learning Techniques

5.4.1 Regression Tree

Regression trees are a methodology for regression analysis. They offer the possibility to achieve good prediction accuracy and easy interpretation, helping us understand the relationship between features and the outcome variable (Yang et al., 2017).
Tree-based models provide great flexibility by enabling us to handle diverse data without the need for extensive pre-processing. These models inherently perform feature selection, eliminating the need for us to select features before starting the modeling process (Kern et al., 2019).
As the name suggests, regression trees utilize a tree-like graph-based model architecture (Yang et al., 2017). The tree is built iteratively by splitting nodes into child nodes. We start with all training samples in one node, the root node, which is split into two child nodes. This is done by identifying one feature and one breakpoint and assigning each sample to one of the two child nodes (Yang et al., 2017). For example, if we use the number of bedrooms as a feature and two bedrooms as the breakpoint, the data is split into two child nodes: one with samples with two or fewer bedrooms and one with samples with more than two bedrooms. The feature and breakpoint, and thus the best split, are identified by testing all possible features and breakpoints and selecting the one that results in the most significant decrease in the residual sum of squares in the case of regression (Kern et al., 2019).
Without predefined stopping criteria, we would end up with huge trees, as the process would only stop when no further decrease in the residual sum of squares is possible. This situation often occurs when there is only one sample in a node. The problem with huge trees is that they perform very well on the training set but poorly on the test set, known as overfitting (Yang et al., 2017).
To mitigate overfitting, stopping criteria are introduced to prevent the construction of excessively large trees. These criteria include, for example, a minimum number of samples per node (Kern et al., 2019). Another approach is pruning, where a large tree is first built, and then branches are cut back. This results in different subtrees and cross-validation is used to find the subtree with the best performance (Kern et al., 2019).

5.4.2 Gradient Boosting Trees

Regression trees are the basis for several advanced model architectures, including Gradient Boosting Decision Trees (GBDT) introduced by Friedman (2001). The key idea behind this model architecture is the concept of boosting, an ensemble technique where multiple weak models are combined to form a single strong model. In the case of GBDT, the weak models are typically decision trees, specifically regression trees for regression problems.
The essential aspect of gradient boosting is the sequential building and combination of multiple weak models. This process aims to minimize the errors of previous trees by sequentially constructing new trees. The goal is to build a new tree that improves the predictions of the previous trees, focusing on the difficult observations that were previously mispredicted (Kern et al., 2019).
When constructing a GBDT for regression, we start with the typical regression problem: We have a set of features and aim to predict a continuous outcome variable. The objective is to estimate or approximate the true relationship or function between the features x and the outcome y, representing a mapping of x to y (Friedman, 2001). This approximation should minimize the expected value of a defined loss function. A key advantage of gradient boosting is its flexibility in using different differentiable loss functions. For regression, the commonly used loss function is the squared error.
The algorithm for building the model can be described in four steps:

1. **Initial Prediction:** Find a constant prediction value γ for all samples that minimizes the defined loss function $L(y_i, \gamma)$ for all samples.

$$F_0(x) = \arg\min_{\gamma} \sum_{i=1}^{n} L(y_i, \gamma)$$

   Building Trees Sequentially: Start a loop where multiple trees are built sequentially, each based on the errors of the previous trees.

2. **Calculating Residuals:** Compute the (pseudo-)residuals r_{im} of the predictions from the existing set of trees. In r_{im} i is the index of the sample, and m is the index of the tree. These residuals are calculated as the negative gradient, which involves computing the derivatives of the loss function with respect to the predictions. For the squared loss function, this equates to the residuals, i.e., the difference between the predicted and true values.

$$r_{im} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x)=F_{m-1}(x)} \quad \text{for } i = 1, \ldots, n$$

3. **Building a New Tree:** Construct a regression tree to predict the (pseudo-)residuals calculated in the previous step. After building the tree, define the values for the leaves such that the prediction value for the residual γ_{jm} in each respective leaf R_{jm} minimizes the loss function. This involves adding the prediction value γ_{jm} for the residual to the prediction value from the previous set of trees $F_{m-1}(x_i)$, considering only the samples in that specific leaf.

$$\gamma_{jm} = \arg\min_{\gamma} \sum_{x_i \in R_{jm}} L(y_i, F_{m-1}(x_i) + \gamma) \quad \text{for } j = 1, \ldots, I_m$$

4. **Updating Predictions:** For all data points, update the prediction values by combining the previous predictions $F_{m-1}(x_i)$ with the predicted residual values γ_{jm} from the new tree. This adjustment is scaled by a learning rate ν to control the extent of correction.

$$F_m(x) = F_{m-1}(x) + \nu \sum_{j=1}^{I_m} \gamma_{jm} 1(x \in R_{jm})$$

After these steps, the process of calculating new residuals and building new trees is repeated iteratively.

5.4.3 K-Nearest Neighbors

As previously described, a central category of supervised learning is classification, which is characterized by the outcome variable that is binary or multi-class (Gupta et al., 2022). Among the numerous algorithms designed to address this issue, the k-nearest neighbor (KNN) algorithm stands out due to its simplicity, effectiveness, and accuracy (Abu Alfeilat et al., 2019). Initially proposed by Evelyn Fix and Joseph Hodges in 1951 and later refined by Thomas Cover and Peter Hart, KNN has been recognized as one of the top ten data mining algorithms, widely studied and applied across various domains (Abu Alfeilat et al., 2019; Guo et al., 2003; Zhang et al., 2017).
The KNN algorithm functions based on a straightforward yet powerful principle: classifying an unlabeled test sample by considering the majority class among its nearest neighbors. The process can be divided into two main phases (Abu Alfeilat et al., 2019; Özöğür-Akyüz et al., 2022; Zhang et al., 2017):

1. **Training Phase**: The algorithm stores the training samples $\{(x_i, y_i)\}_{i=1}^{N}$ where $x_i \in R^d$ is the feature vector and $y_i \in \{1, \ldots, C\}$ is the corresponding class label. There must be no missing or non-numeric data.

2. **Classification Phase**:

   (a) **Distance Calculation**: For each test sample x_{test}, the algorithm calculates the distances to all stored training samples using a specific distance function or similarity measure $d(x_{\text{test}}, x_i)$.

   (b) **Neighbor Selection**: The k-nearest neighbors are selected, where k is a predefined small integer. These are the training samples with the smallest distances to x_{test}.

   (c) **Class Assignment**: The test sample x_{test} is assigned the most frequent class among these k neighbors. Formally, let $N_k(x_{\text{test}})$ denote the set of the k nearest neighbors of x_{test}. The predicted class $\hat{y}_{\text{test}}$ is given by:

$$\hat{y}_{\text{test}} = \arg \max_{y \in \{1,\dots,C\}} \sum_{i \in N_k(x_{\text{test}})} I(y_i = y)$$

   where I is the indicator function, which equals one if the argument is true and zero otherwise.

The success of KNN heavily depends on the choice of k and the distance measure used. Common distance measures include the Euclidean distance for real-valued features and the Hamming distance for binary features.
Despite its simplicity, KNN has some notable disadvantages, primarily related to its computational and memory requirements (Abu Alfeilat et al., 2019):

1. **Computational Cost**: For each test sample, the algorithm computes the distance to all training samples, resulting in a time complexity of $O(nm)$, where n is the number of training samples and m is the number of features per sample.

2. **Memory Requirement**: The algorithm needs to store all training samples, leading to a space complexity of $O(nm)$.

The value of k significantly influences the performance of KNN. An inappropriate choice can bias the classification results. The optimal k value is typically determined through experimentation, where the algorithm is run multiple times with different k values, and the one yielding the best performance is selected (Guo et al., 2003).
Overall, KNN is a nonparametric, instance-based learning algorithm that leverages the entire training dataset to make predictions at the classification stage.

5.5 Feature Extraction Methods

5.5.1 Principal Component Analysis

Principal Component Analysis (PCA) is a fundamental technique in modern data analysis. It is a non-parametric method designed to simplify complex datasets by transforming them into lower-dimensional forms, revealing underlying structures that might not be immediately apparent. In dimensionality reduction, PCA aims to maximize the variance retained in the dataset, filter out noise, and highlight hidden structures within the data. This reduction is achieved by identifying principal components, which are directions in the data that exhibit maximum variation (Kherif & Latypova, 2020; J. Li et al., 2022; Shlens, 2014).
The process involves transforming the original dataset X into a new dataset Y through a transformation matrix P. Geometrically, P acts as both a rotation and a stretch, reorienting and rescaling X to achieve Y. As described, the goal is to capture as much variance as possible in the data X through the principal components, which are contained in our transformation matrix P (J. Li et al., 2022). We can obtain the necessary transformation through the eigendecomposition and the resulting eigenvectors of the covariance matrix of our original dataset X, such that the following holds:

$$Cv_i = \lambda_i v_i$$

Where C is the variance-covariance matrix, v_i is one eigenvector (there are multiple eigenvectors), and λ_i is the corresponding eigenvalue. The resulting eigenvectors from the eigendecomposition of the variance-covariance matrix are called principal components. The principal components provide the directions in which the data varies the most. The corresponding eigenvalues indicate the magnitude of the variance in these directions. Finally, these principal components are then used to transform our original data X and reduce its dimensionality (Kherif & Latypova, 2020; Shlens, 2014).

PCA operates under several key assumptions, which can impact its performance (Shlens, 2014):

1. **Linearity**: PCA assumes the data can be re-expressed as a linear combination defined through the principal components.

2. **Large Variances Represent Important Structures**: This assumption implies that components with larger variances are more significant and represent meaningful structures in the data, whereas components with smaller variances are considered noise.

3. **Orthogonality of Principal Components**: PCA assumes that the principal components are orthogonal to each other. This assumption simplifies the mathematical computations of PCA, making it solvable using linear algebra techniques.

A common method to obtain the principal components for PCA is Singular Value Decomposition (SVD).
SVD decomposes any $m \times n$ matrix X into three matrices:

$$X = U\Sigma V^T$$

Where U is an $m \times m$ unitary matrix, Σ is an $m \times n$ diagonal matrix with non-negative real numbers on the diagonal (singular values), and V is an $n \times n$ unitary matrix.
Regarding PCA, the singular values in Σ represent the variance captured by each principal component and correspond to the square roots of the eigenvalues. The columns of U represent the eigenvectors or principal components of X, whereas the columns of V represent the eigenvectors or principal components of X^T (J. Li et al., 2022; Shlens, 2014).

Performing PCA can overall be summed up into the following five steps:

1. **Organize Data**: Arrange the dataset X as an $m \times n$ matrix, where n is the number of features and m is the number of observations.

2. **Mean Subtraction**: Subtract the mean of each feature from the dataset.

3. **Compute Covariance Matrix**: Calculate the covariance matrix of the mean-subtracted data.

4. **Eigendecomposition or SVD**: Perform SVD on the covariance matrix to obtain the eigenvectors (principal components) and eigenvalues.

5. **Projection**: Project the original data onto the space spanned by the principal components.

5.5.2 Clustering

Clustering is a fundamental technique in data analysis and machine learning used to partition a dataset with different data points into groups or clusters that contain similar data points. The goal is to ensure that data points within the same cluster are more similar to each other than to those in other clusters. This

similarity is often measured using distance metrics, such as Euclidean distance (Mehrotra et al., 1997).
K-means clustering is one of the most widely utilized clustering methods. K-means seeks to partition a set of n data points in R^d (a d-dimensional real space) into k clusters by minimizing the mean squared distance from each data point to its nearest cluster center, also known as squared-error distortion (Alsabti et al., 2000; Kodinariya & Makwana, 2013).
The k-means algorithm, a popular heuristic for solving the k-means clustering problem, operates through a straightforward iterative process aimed at finding a locally optimal solution. The procedure can be outlined as follows (Alsabti et al., 2000; Kodinariya & Makwana, 2013):

1. **Initialization**: Select k initial centroids c_j with $j \in \{1, \ldots, k\}$, ideally spaced as far apart as possible to improve the chances of convergence to a good solution.

2. **Assignment**: Assign each data point x_i with $i \in \{1, \ldots, n\}$ to the nearest centroid, forming k clusters C_j $j \in \{1, \ldots, k\}$.

3. **Update**: Recalculate the centroids of the k clusters as the mean of the data points assigned to each cluster $c_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} x_i$.

4. **Iteration**: Repeat the assignment and update steps until the centroids no longer change their positions.

The objective function minimized by this algorithm is the sum of squared distances between each data point and its assigned centroid. The process can be described as follows:

$$\sum_{i=1}^{n} \min_{1 \leq j \leq k} \|x_i - c_j\|^2$$

where x_i represents a data point and c_j denotes a centroid.
Determining the optimal number of clusters, k, is critical to k-means clustering. Various methods have been proposed to address this issue (Kodinariya & Makwana, 2013):

1. **Rule of Thumb**: A heuristic approach that can be applied to any dataset.
2. **Elbow Method**: Plots the explained variance as a function of k and identifies the "elbow point" where the reduction of the squared-error distortion starts to slow down.
3. **Information Theoretic Approach**: Applies principles from information theory to find the optimal number of clusters.

4. **Silhouette Method**: Measures how similar a data point is to its own cluster compared to other clusters.
5. **Cross-Validation**: Utilizes resampling techniques to assess the stability and reliability of the chosen k.

The k-means Algorithm can be computationally demanding, especially for large datasets with many dimensions. The computational complexity per iteration includes the cost of assigning n data points to the nearest centroid and updating the k centroids. The number of iterations required can vary significantly depending on the dataset's characteristics, ranging from a few to several thousand iterations (Alsabti et al., 2000).

5.6 Feature Importance

5.6.1 SHAP

5.6.1.1 Shapely Values

SHAP (SHapley Additive exPlanations), introduced by Lundberg and Lee (2017), is a method rooted in game theory and based on shapley values. Shapley values aim to fairly distribute a "payout" among players in a game theory setting based on their contributions. In a machine learning context, each feature in a dataset can be seen as a "player," and the model's prediction is the "payout." Shapley values quantify the contribution of each feature to the model's prediction.
Since SHAP builds upon shapley values, it is crucial to understand their computation. Shapley values determine the average contribution of a feature to the prediction of all possible combinations of features (coalition). This ensures that the effect of one feature on the model's predictions is defined through all possible feature combinations and their respective predictions.
Mathematically, the shapley value $\phi(j)$ for feature j is the average marginal contribution of that feature across all possible coalitions of features. To make this clearer, we define an exemplary scenario. Consider a model with three features: *Age, Square Feet, and Year Sold.* For a given instance, we predict using only one feature, *Age*, and get an outcome (house price) $\hat{Y}_{Age} = 100,000$. If we take the average of all house prices, denoted as $\hat{Y}_{\emptyset} = 80,000$, the marginal contribution is $20,000$, which is the difference between the prediction including no features $\hat{Y}_{\emptyset}$ and the prediction of a model $\hat{Y}_{Age}$ including the feature *Age*.

$$MC_{Age,\{Age\}}(x_i) = \hat{Y}_{Age}(x_i) - \hat{Y}_{\emptyset}(x_i) = \$100K - \$80K = \$20K$$

Since the Shapley value $\phi(j)$ of the feature *Age* is the average marginal contribution across all possible coalitions, for each coalition, the marginal contribution

of the feature is calculated as the difference in the prediction with and without the feature. The shapley value is then the weighted average of these marginal contributions.
In our example, we would consider all possible combinations of features. We can combine the feature Age with the following feature combinations:

1. *No variables*
2. *Square Feet*
3. *Year Sold*
4. *Square Feet & Year Sold*

For each coalition, the predicted outcome is computed with and without the feature Age, and the difference represents the marginal contribution of Age. Note that the marginal contribution is calculated on the instance level, meaning we select specific data points from our dataset to calculate the outcomes and, thus, the marginal distributions. The shapley value $\phi(j)$ is the average of these marginal contributions.
The following formula summarizes the calculation of shapley values:

$$\phi_j = \sum_{S \subseteq F\{j\}} \frac{S!(F-S-1)!}{F!} \left[g_x(S \cup j) - g_x(S)\right]$$

$$\text{where } g_x(S) = E[g(x)|x_S]$$

Here, S represents a subset of the features, F denotes the set of all features, and $E[g(x)|x_S]$ represents the expected value of the model on the subset S (Lundberg & Lee, 2017). The first factor within the sum is used as a weighting factor for the marginal contributions.
Most models are unable to process missing values in the input data. Therefore, when a variable is not included in a coalition, a value is randomly selected from the training data to represent the average for that variable. To enhance the reliability of the estimates for the shapely values, the process of sampling and computing marginal contributions is repeated multiple times, and the results are averaged at the end (Lundberg & Lee, 2017).

5.6.1.2 Kernel SHAP

Lundberg and Lee (2017) proposed SHAP as a new method to estimate shapley values. This method is called the kernel SHAP, which essentially is a weighted linear regression with a choice of kernel function to weigh different data points.
It uses a linear explanatory model of shapley values to estimate the initial prediction model:

$$\hat{h}(z) = \phi_0 + \sum_{i=1}^{M} \phi_i z_i$$

In this equation, $h(z)$ represents the output of the explanation model, i.e., the model for which we want to examine the importance of different features, whereas z defines the coalition, indicating which features are included and which are not. Thus, z only includes values of either 0 or 1, $z \in \{0,1\}^M$. M is the maximum size of the coalition, and ϕ denotes the feature attribution, or the shapley value (Lundberg & Lee, 2017).

To solve the linear regression problem, a synthetic dataset for the sample of interest is computed using different vectors z (Molnar, 2020). The dataset then consists of the coalition vectors and their corresponding model estimates $h(z)$. As we aim to minimize the variance of the shapley value estimates, for each coalition vector, we sample M times (unless z is a vector of 1s). With this new dataset, we fit the simple linear model in the equation above. The shapley values are the model's coefficients, typically estimated by kernel methods (Molnar, 2020).

SHAP allows for both local and global model interpretations. Two methods for global interpretation are:

1. *SHAP Variable Importance*: Measures the average absolute effect of each variable by averaging the absolute shapley values across all data points (Molnar, 2020).

$$I_{SHAP}(j) = \frac{1}{N} \sum_{i=1}^{N} |\phi_j^i|$$

2. *SHAP Summary Plot*: Visualizes shapley values for all variables and samples, showing the distribution of contributions and providing an interpretable representation of feature impacts (Molnar, 2020).

In the context of house price prediction and value appreciation prediction, SHAP values can identify the contribution of various features to the predicted prices and their changes over time. SHAP provides a framework for evaluating and interpreting our models and enables local and global insights into our model predictions.

5.6.2 Permutation Importance

Permutation importance is a technique to assess the significance of features in a predictive model. The initial idea was introduced by Breiman (2001) in the context of Random Forest Trees. Building upon Breimann's idea Fisher et al. (2019) developed a model-independent version of permutation importance. The method compares the baseline performance of the model against the performance

when the data is permuted.
Therefore, it measures the increase in the model's prediction error after permuting the feature's values. This disruption breaks any relationship between the feature and the true outcome, thereby helping to identify how crucial that feature is for the model's predictive accuracy.
To elaborate, consider a supervised learning task where a model f is trained on a dataset X with labels y, and its performance is evaluated using a loss function L. Permutation importance evaluates how much the loss L increases when a specific feature X_i is randomly shuffled, thereby isolating its effect on the model's performance.
As introduced by Fisher et al. (2019) computing permutation importance includes the following four steps (Molnar, 2020):

1. **Train the Model**: Train the model f on the dataset X and calculate the baseline error $e_{orig} = L(y, f(X))$.

2. **Permute Feature Values**: For each feature j, create a permuted version of the dataset X_{perm} by randomly shuffling the values of feature j. This shuffling breaks the association between feature j and the true outcome y.

3. **Calculate Permuted Error**: Evaluate the model on the permuted dataset to obtain the new error $e_{perm} = L(y, f(X_{perm}))$.

4. **Compute Importance Score**: The importance score for feature j can be computed as either the quotient $FI_j = \frac{e_{perm}}{e_{orig}}$ or the difference $FI_j = e_{perm} - e_{orig}$.

A feature is deemed "important" if permuting its values substantially increases the model error, indicating that the model relied on this feature for making accurate predictions. Conversely, if the error remains essentially unchanged, the feature is considered "unimportant".
Permutation importance has several advantages. It does not require prior knowledge of feature distributions, making it robust across various datasets and models. Moreover, it is model-agnostic and applicable to any predictive model, whether linear regression, decision trees, or neural networks.
However, multicollinearity can influence permutation importance, where features are highly correlated. In such cases, permuting one feature may not significantly affect the model's performance if other correlated features still capture its information. Some approaches mitigate this problem, such as by permitting correlated features together. However, this approach requires defining correlation thresholds and can be computationally intensive (Pereira et al., 2021).

5.7 Model Evaluation Metrics

Various metrics are available to evaluate regression models' performance regarding their accuracy in predicting continuous outcomes. This section introduces some of the most commonly used evaluation metrics for regression models, explaining their calculations and interpretations. Understanding these metrics helps us select the suitable model for our prediction problems and fine-tune them to achieve optimal results.

1. Mean Squared Error (MSE)
The Mean Squared Error (MSE) is a key metric used to evaluate regression models. It is part of a group of metrics that assess the distance between the model's predictions and the true values, helping to measure the accuracy of those predictions. Specifically, it measures the average of the squares of the errors, which are the differences between the actual and predicted values (Sammut & Webb, 2011). Mathematically, it is represented as:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

where Y_i denotes the actual value, $\hat{Y}_i$ denotes the predicted value, and n is the number of observations. MSE is sensitive to outliers due to the squaring of errors, making it useful in scenarios where large errors are particularly undesirable. Lower MSE values indicate better model performance.

2. Root Mean Squared Error (RMSE)
Root Mean Squared Error (RMSE) is the square root of MSE, providing an error metric in the same units as the original data, which aids interpretability (Chicco et al., 2021). The formula for RMSE is:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}$$

Similar to MSE, RMSE penalizes more significant errors more heavily and is often used in practical applications due to its intuitive interpretation.

3. Mean Absolute Error (MAE)
Mean Absolute Error (MAE) measures the average of the errors in a set of predictions without considering their direction (only considering the absolute errors) (Fürnkranz et al., 2010). It is calculated as:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i|$$

MAE is less sensitive to outliers than MSE and RMSE, as it does not square the errors. This makes MAE a useful metric when the model needs to perform consistently across all observations without giving more weight to outliers.

4. Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) expresses prediction accuracy as a percentage, making it easier to understand the model's performance in relative terms (de Myttenaere et al., 2016). It is defined as:

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{Y_i - \hat{Y}_i}{Y_i}\right| \times 100$$

MAPE is particularly useful in applications where the scale of the data varies. Still, it can be problematic if the actual values Y_i are close to zero, leading to significant percentage errors.

5. Symmetric Mean Absolute Percentage Error (SMAPE)

Symmetric Mean Absolute Percentage Error (SMAPE) is a variation of MAPE that normalizes the absolute error by the average of the actual and predicted values, aiming to address some of MAPE's limitations (Chicco et al., 2021). It is given by:

$$\text{SMAPE} = \frac{100\%}{n}\sum_{i=1}^{n}\frac{|Y_i - \hat{Y}_i|}{(|Y_i| + |\hat{Y}_i|)/2}$$

SMAPE ranges from 0% to 200%, with 0% indicating a perfect fit. It is less biased towards low forecasts and provides a more balanced view of model performance, especially in datasets with both small and large values (Chicco et al., 2021).

6. Coefficient of Determination (R-squared)

The Coefficient of Determination, R^2, introduced by Wright (1921), indicates the proportion of the variance in the dependent variable that is predictable from the independent variables. It is calculated as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}$$

where $\bar{Y}$ is the mean of the observed data. R^2 values range from 0 to 1, with 1

indicating a perfect fit. Negative values can occur if the model is worse than a horizontal line representing the mean of the dependent variable.
Each of these metrics provides different insights into the performance of regression models. MSE and RMSE are sensitive to outliers, while MAE offers a straightforward average error measurement. MAPE and SMAPE provide relative error measures, with SMAPE addressing some of MAPE's limitations. R^2 indicates the proportion of variance explained by the model but should be combined with other metrics for a comprehensive evaluation.

6 Related Work

Given that the business model presented and tested in this paper is novel, there is no existing research directly related to it. However, the process of testing this business model intersects with certain subfields. Specifically, the business model requires using two machine learning models: one for accurately predicting house prices and another for predicting house price appreciation. This section will examine these two research fields and review the existing literature.

6.1 House Price Prediction

House price prediction has been a significant research area since the 1970s, primarily driven by economic theories and statistical methods. One of the foundational models in this domain is the hedonic price model, introduced by Rosen (1974). This model employs regression methods to examine the relationship between house prices and various house features. The hedonic price model has been extensively studied and applied in the literature, including works by Can (1992), Król (2015), and Yayar and Demir (2015), demonstrating its widespread applicability in different contexts. Despite its robustness in analyzing feature-price relationships, the hedonic model is often limited by its reliance on linear assumptions. It does not account for more complex, non-linear interactions between features. The hedonic model also struggles with problems like outliers, discontinuity, and fuzziness (Özöğür-Akyüz et al., 2022). The original hedonic model by Rosen (1974) considered only the characteristics of the house, ignoring other external factors. However, the model has since been modified to include additional external factors in determining house prices (Y. Li et al., 2023). Techniques from spatial econometrics were used to extend the original hedonic model, to also consider spatial effects (Geerts et al., 2023).

Numerous studies have underscored the importance of location and surrounding community features in determining house prices. Research by Bourassa et al. (2010) and Case et al. (2004) emphasized the spatial dependencies and temporal patterns in house price predictions, advocating for models that consider these aspects. These studies typically approach the problem by partitioning housing data based on geographic or community characteristics and developing localized prediction models. This approach allows for a more granular analysis of how location-specific factors influence house prices. L. Wang et al. (2022) also considered spatiotemporal characteristics in one model. They used spatial density and distance features to consider different points of interest.

In recent years, the application of machine learning techniques to house price

prediction has gained significant attention. Machine learning models such as Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), and Adaptive Boosting (AdaBoost) have shown promising results in capturing the complex, non-linear relationships inherent in housing data. For instance, Selim (2009) compared hedonic regression with ANNs for predicting house prices in Turkey, finding that ANNs provided more accurate predictions. Similarly, studies by Gu et al. (2011) and X. Wang et al. (2014) have highlighted the effectiveness of SVMs. Park and Bae (2015) also demonstrated the utility of AdaBoost in enhancing prediction accuracy.

As another Tree-based model, Random Forests have been shown to outperform traditional regression models by better capturing the non-linearity in housing data. Koktashev et al. (2019) and Wu and Wang (2018) demonstrated the superior performance of Random Forests in predicting house prices in various urban contexts.

With the advent of big data and advanced computational techniques, more sophisticated models have been proposed for predicting house prices. Zhao et al. (2019) integrated a Convolutional Neural Network (CNN) with XGBoost to enhance real estate appraisal accuracy, highlighting the potential of deep learning in this field.

Özöğür-Akyüz et al. (2022) introduced a hybrid model for house price prediction by combining both hedonic regression and machine learning techniques, such as methods like clustering and classification. Recent approaches have also focused on leveraging multi-modal data, including textual and image data, to improve house price predictions. Abdallah (2015) proposed a two-stage model that combines structured numerical features with textual features extracted from real estate listings using text mining techniques. This integration of diverse data sources has been shown to enhance the predictive performance of house price models. Moreover, using natural language processing (NLP) and transformer models to analyze text data, as explored by various studies, represents a novel direction in this research area (Y. Li et al., 2023).

The evolution of house price prediction models reflects a shift from traditional hedonic and linear regression methods to more advanced machine learning and deep learning techniques. These modern approaches offer improved accuracy and the ability to handle complex, non-linear relationships in housing data. Integrating location-based factors and multi-modal data further enhances these models' robustness and overall performance.

6.2 House Price Appreciation Prediction

House price appreciation prediction remains a relatively unexplored domain despite the significant interest and research directed towards house price prediction. The primary distinction between these two research areas is that while house price modeling provides a snapshot of property values at a given point in time, house price appreciation prediction offers insights into the growth or decay of property values over an extended period. This differentiation is crucial, as high property prices do not necessarily correlate with high appreciation rates, which can be influenced by different variables (Kang et al., 2021).

One of the most noteworthy studies in this area is the paper titled "Understanding House Price Appreciation Using Multi-Source Big Geo-Data and Machine Learning" by Kang et al. (2021). Their research underscores the importance of evaluating house price appreciation to support place-based decision-making and real estate market analyses. The authors used an extensive dataset consisting of house structural attributes, house photos, locational amenities, street view images, transportation accessibility, visitor patterns, and socioeconomic attributes of neighborhoods.

Their study involved analyzing over 20,000 houses in the Greater Boston Area to explore the spatial dependency of house price appreciations, influential variables, their relationships and overall predictions by applying and combining multiple machine learning models such as ResNet, Gradient Boosting Trees, as well as multiple linear regression.

Another paper in the field is the paper by Carrillo et al. (2014) "Can Tightness in the Housing Market Help Predict Subsequent Home Price Appreciation? Evidence from the U.S. and the Netherlands". This study assesses the predictive power of market tightness indicators on future house prices, such as seller's bargaining power and sale probabilities. The authors focus on the theoretical connections between these indicators and subsequent house price appreciation. The empirical analysis utilizes listings data from residential units in the Netherlands and specific U.S. regions. The findings reveal that these indicators can significantly reduce house price appreciation prediction errors. Through their work they demonstrate how listings data can be leveraged to predict house price appreciation and overall how the data can be used to get insight about the future state of the housing market.

However, the focus of this paper lies more on the economic factors and their effect on the housing market and house price appreciation, and less on prediction performance and the application of novel machine learning methods.

There is very little research done that aims to predict house price appreciation.

Several studies analyze house price appreciation without developing predictive models. For example, the paper "Local Market and National Components in House Price Appreciation" by Gyourko and Voith (1992) uses a long time series and large cross-section to provide insights into investment opportunities across metro areas. Another study, "Neighborhood Diversity and House-Price Appreciation" by Macpherson and Sirmans (2001), examines the relationship between house price appreciation and changes in racial/ethnic composition in certain Florida counties.

Overall, research on house price appreciation prediction is limited. While many papers analyze house price appreciation alongside other economic factors, they often do not develop predictive models. Existing studies emphasize the importance of using multi-source data, machine learning, and market tightness indicators to enhance the understanding and prediction of house price appreciation trends.

7 Results

7.1 Data Analysis House Price Prediction

7.1.1 Correlation

It is crucial to examine the correlation coefficients to gain an initial understanding of the relationship between various features and the target variable, which in this case is the house price. This analysis also helps to understand the interrelationships among the features themselves. However, it is important to note that correlation coefficients only measure and capture linear relationships between two random variables. This limitation must be considered when interpreting the results of such analyses.

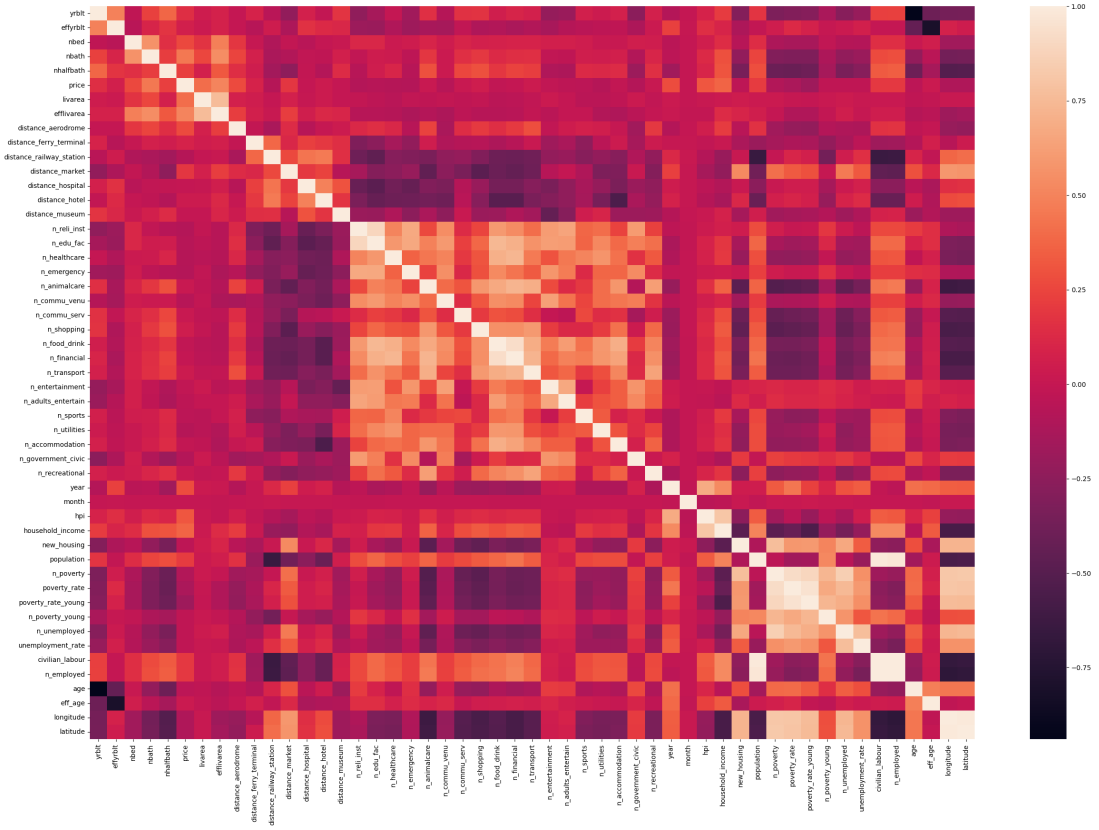


Figure 5: Correlations

7.1.1.1 Geographic Feature Correlations

Our analysis reveals a positive correlation among most geographic frequency features. For example, features indicating the number of facilities or amenities within a certain radius tend to correlate positively (see Figure 5). Conversely, there is a predominantly negative correlation between geographic distance and frequency

features. This relationship makes sense; for instance, a high number of healthcare facilities within a 5 km radius likely implies a shorter distance to the nearest hospital, which is included in the count of healthcare facilities.

7.1.1.2 Economic Factors and Population

Economic factors exhibit substantial positive correlations. For example, there is a significant positive correlation between population size and the number of employed persons. This is expected, as larger cities or counties with higher populations typically have a larger total number of employed individuals compared to regions with smaller populations. However, the relationship is weak when examining the correlation between population and unemployment rate. This is because the unemployment rate is a relative measure, varying independently of the population size.

7.1.1.3 Negative Correlations in Economic Factors

Among the economic indicators, some features are negatively correlated. For instance, there is a negative correlation between the poverty rate and the number of employed persons. This makes economic sense, as higher poverty rates generally indicate fewer employed individuals, reflecting a weaker economic condition. Similarly, there is a negative correlation between the unemployment rate and household income. A high unemployment rate suggests economic distress, which is likely associated with lower average household incomes.

7.1.1.4 Housing Market Indicators

A particularly interesting finding is the strong positive correlation between the house price index and the number of new housing permits. Since the house price index reflects the price development of houses, an increase in house prices is likely to coincide with increased interest in building or acquiring new houses.

7.1.1.5 Population and Geographic Distance Features

Another noteworthy observation is the significant correlation between demographic factors, such as population, and certain geographic distance features. For instance, there is a strong negative correlation between population size and the distance to the nearest railway station. This likely occurs because higher-population areas tend to have more railway stations, resulting in shorter distances to these stations compared to less populated regions. A similar trend is observed with the number of employed individuals, which also shows a strong positive correlation with population size. Thus, these results suggest that areas

with larger populations and higher employment rates typically have better access to transportation infrastructure.

7.1.1.6 Correlation with Sale Price

Several observations can be made when examining the correlations between various features and the target variable, house price. No feature exhibits a significantly high correlation with house price.

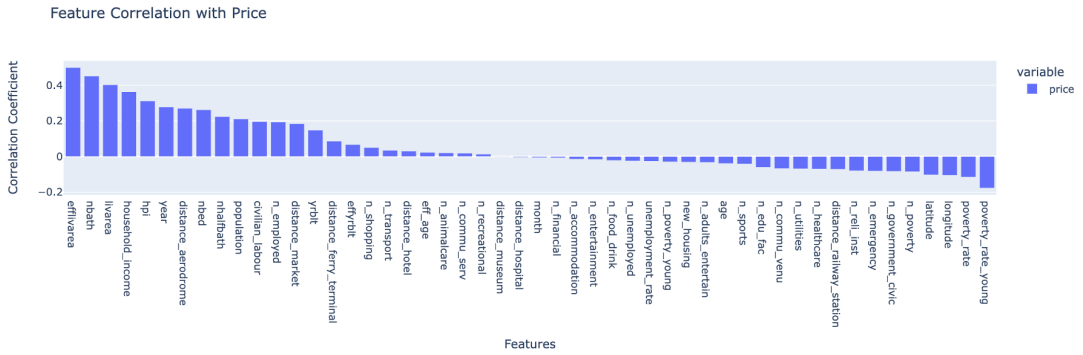


Figure 6: Correlation with Sale Price

The feature displaying the highest correlation with house price is the effective living area, with a correlation coefficient of 0.50 (see Figure 6). The effective living area includes the above-grade living space and the size of the basement rooms but excludes areas such as the garage square footage. While this correlation is the highest among the features considered, it still indicates a relatively weak relationship. Among economic factors, following the living area and the number of bathrooms, household income, and the house price index show notable correlations with the house price. Although not exceedingly high, these correlations suggest that economic conditions play a role in determining house prices. Regarding geographical distance features, the distance to the airport shows the highest correlation with the house price. This indicates that proximity to airports may have a tangible impact on property values. Interestingly, geographical frequency features exhibit very low correlations with the house price. This finding suggests that the frequency of certain amenities or facilities within a certain radius does not significantly affect house prices. However, as stated before, these features might not have a clear linear relationship with the target variable; instead, they could have a more complex, non-linear relationship. In this case, they would still have a low correlation coefficient, as it only captures linear relationships.
In summary, while certain features such as the (effective) living area, number of bathrooms, household income, and proximity to airports show some degree of correlation with the house price, no single feature stands out as having a strong

influence. This highlights the complexity of factors affecting house prices and suggests that a multifaceted approach is necessary to comprehensively understand property value determinants.

7.1.2 Descriptive Statistics

To gain a better insight into our dataset and its general structure, we calculated and analysed descriptive statistics for each feature (see Table 4). This section presents some of the key findings.

Regarding location, most houses are approximately 17 to 40 kilometers away from the nearest ferry terminal and around 6 to 13 kilometers from the nearest airport. Regarding their condition, about 60% of the houses are in average condition, while approximately 26% are in good condition.

In terms of the number of transactions in our dataset, the cities of Alexandria, Fairfax, and Springfield account for the majority, whereas Plainville, Darien, and Arlington have the fewest. At the county level, most transactions occur in Fairfax, Hartford, and New Haven, whereas Windham, Middlesex, and Tolland experience the fewest.

The typical house has three bedrooms, although there are outliers, such as a hotel with 124 bedrooms. A similar picture can be seen for the number of bathrooms. Most houses have two bathrooms; however, a notable outlier is a real estate agency building with 239 bathrooms.

Most houses were sold immediately after construction, with an age ranging from 0 to 1 year. Overall, the houses sold range in age from newly built (0 years) to 116 years old. The effective age of most houses at the time of sale is between 18 and 27 years.

Regarding construction years, most houses were built in 1986, with the majority constructed between 1956 and 1988. The refurbishment years for most houses range from 1987 to 2001, with a concentration around 1992.

	mean	std	min	25%	50%	75%	max
yrblt	1968.28	31.61	1500.00	1956.00	1974.00	1988.00	2023.00
effyrblt	1993.46	12.81	1697.00	1987.00	1994.00	2001.00	2023.00
nbed	3.48	1.08	0.00	3.00	3.00	4.00	124.00
nbath	2.25	1.05	0.00	2.00	2.00	3.00	239.00
nhalfbath	0.80	0.60	0.00	0.00	1.00	1.00	20.00
price	396981.25	459674.77	2.00	172000.00	285000.00	478000.00	13143478.00
livarea	2069.78	2356.39	0.00	1317.00	1702.00	2420.00	456237.00
efflivarea	2309.12	1410.26	0.00	1521.00	2009.00	2814.00	347618.00
distance_aerodrome	10469.93	5310.74	6.94	6753.57	9732.71	13333.20	32311.00
distance_ferry_terminal	29401.70	15504.18	82.30	17093.86	27907.65	41912.91	106140.25
distance_railway_station	8073.89	8645.22	56.18	2850.81	5499.12	9991.78	83584.58
distance_market	12979.15	14285.89	1.25	3156.32	6659.00	18644.46	79067.72
distance_hospital	7221.87	4799.68	5.42	3751.16	6235.52	9641.51	39881.43
distance_hotel	4632.80	3591.32	0.00	2103.82	3540.69	6095.59	26541.72
distance_museum	4979.98	2829.72	0.00	2797.36	4559.21	6787.82	24473.30
n_reli_inst	22.11	19.23	0.00	9.00	18.00	31.00	128.00
n_edu_fac	25.11	17.47	0.00	11.00	23.00	35.00	126.00
n_healthcare	8.94	8.75	0.00	2.00	7.00	13.00	47.00
n_emergency	5.11	3.27	0.00	3.00	4.00	7.00	23.00
n_animalcare	10.55	11.99	0.00	2.00	6.00	15.00	65.00
n_commu_venu	4.60	3.96	0.00	2.00	4.00	6.00	39.00
n_commu_serv	0.60	0.97	0.00	0.00	0.00	1.00	5.00
n_shopping	0.82	1.25	0.00	0.00	0.00	1.00	6.00
n_food_drink	63.59	57.08	0.00	18.00	51.00	94.00	326.00
n_financial	11.00	9.24	0.00	4.00	9.00	16.00	46.00
n_transport	28.67	37.19	0.00	2.00	13.00	42.00	181.00
n_entertainment	3.98	4.38	0.00	1.00	3.00	5.00	39.00
n_adults_entertain	2.45	5.33	0.00	0.00	1.00	3.00	51.00
n_sports	0.90	1.32	0.00	0.00	0.00	1.00	10.00
n_utilities	6.78	7.81	0.00	1.00	4.00	10.00	49.00
n_accommodation	4.34	5.24	0.00	0.00	2.00	7.00	25.00
n_government_civic	0.41	1.22	0.00	0.00	0.00	0.00	8.00
n_recreational	6.78	9.68	0.00	1.00	3.00	8.00	60.00
year	2007.15	11.86	1900.00	2000.00	2008.00	2018.00	2023.00
month	6.84	3.17	1.00	4.00	7.00	9.00	12.00
hpi	141.68	43.60	74.92	113.81	136.61	162.16	276.86
household_income	79316.03	22337.71	36770.00	62607.00	75094.00	91170.00	144632.00
new_housing	763.22	819.15	5.00	52.00	456.00	1292.00	3119.00
population	870.68	246.99	102.76	857.85	897.62	989.31	1152.36
n_poverty	41716.35	42719.50	861.00	1460.00	14015.00	86299.00	112801.00
poverty_rate	7.84	2.57	4.10	5.70	7.10	10.30	13.60
poverty_rate_young	10.46	4.13	4.50	6.70	9.40	13.80	20.70
n_poverty_young	20801.07	8129.02	1384.00	16370.00	20596.00	26492.00	42079.00
n_unemployed	11038.90	12702.02	73.00	345.00	5310.00	20133.00	61790.00
unemployment_rate	4.44	2.16	1.10	2.90	4.00	5.30	17.30
civilian_labour	475004.85	139574.84	54991.00	453736.00	484193.00	558684.00	673346.00
n_employed	454750.51	136311.10	50472.00	424247.00	463636.00	543457.00	657809.00
age	38.88	34.62	0.00	14.00	32.00	55.00	518.00
eff_age	19.56	13.54	0.00	10.00	18.00	27.00	314.00
longitude	-74.98	2.20	-77.53	-77.26	-73.52	-72.90	-71.80
latitude	40.24	1.34	38.64	38.84	41.13	41.53	42.05

Table 4: Descriptive Statistics

Variable	Unique	Top	Frequency
city	149	Alexandria	74900
cond_desc	5	Average	610260
saledate	15842	1966-12-31	761
county	9	Fairfax	477166
state	2	Connecticut	524842

Table 5: Summary of Categorical Values

7.1.3 Socioeconomic Features

The analysis of socioeconomic indicators across the different counties reveals distinct patterns and disparities. Poverty rates in New Haven, Hartford, and Windham counties have consistently been much higher compared to other counties over the years, indicating continuous socioeconomic challenges in these areas (see Figure 7). In contrast, Fairfield County shows a more balanced economic status in this regard, lying in the middle area compared to the other counties.
Population trends show that Fairfax County maintains the highest population and demonstrates the most substantial growth over the years (see Figure 7). This indicates a robust and expanding population base. Fairfield, Hartford, and New Haven counties also have high population levels, each exceeding 800,000 residents, highlighting a concentration of population in these urban areas. With around 200,000 residents, the other counties have significantly lower populations.

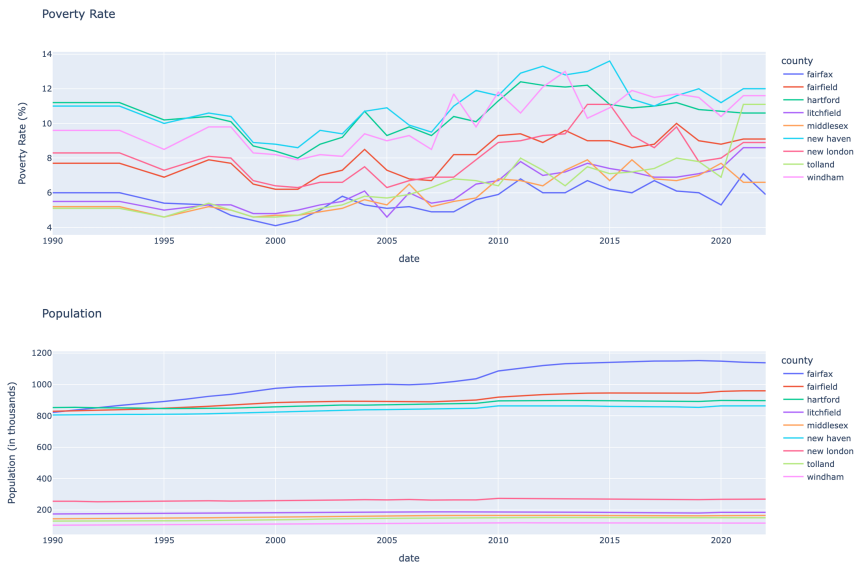


Figure 7: Poverty Rates and Population Development

An interesting aspect of the analysis is the number of new housing permits issued (see Figure 8). Despite having the highest population and a low poverty rate, Fairfax County exhibits a surprisingly low number of new housing permits. This suggests potential constraints or challenges in housing development within the county. Around 2009 and 2010, the number of new housing permits reached its lowest point across all counties, likely reflecting the impacts of the economic recession during that period. Fairfield County experienced a significant increase in new housing permits in 2011, surpassing the levels seen in other counties, which indicates an active housing market. New Haven and Hartford counties also show high levels of new housing permits, aligning with their larger populations.

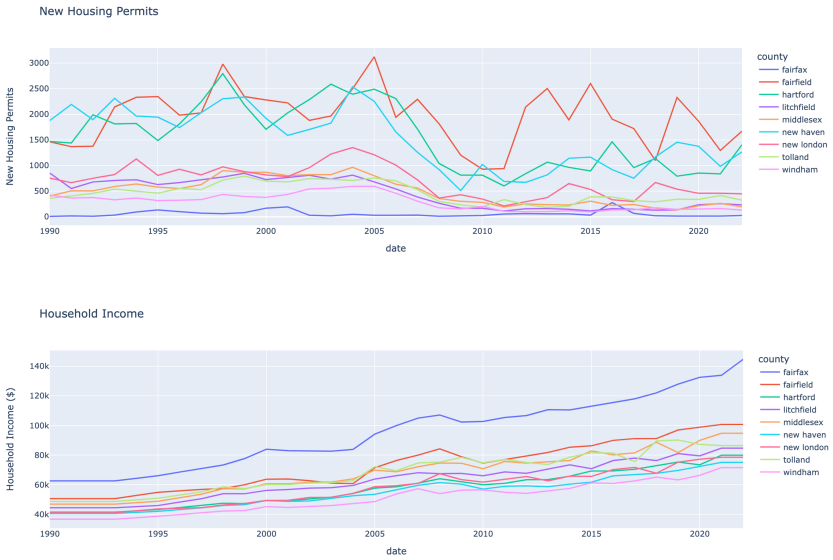


Figure 8: New Housing Permits and Household Income

Household income analysis highlights a contrast between the counties. Fairfax County has significantly higher household incomes, emphasizing its economic prosperity (see Figure 8). In comparison, Windham and New Haven counties have the lowest household incomes, reflecting the economic challenges in these areas. The income gap in 2022 between Fairfax ($144,000) and the second-highest Fairfield ($100,000) underscores with a difference of $40,000 a high disparity.

The house price index (HPI) presents an evolving picture of the housing market. Before 2000, the HPI was relatively consistent across counties. However, post-2000, Fairfax County experienced a much more substantial rise in the HPI compared to the other counties. The overall trend reveals a peak in the HPI in 2006, followed by a significant decline during the economic downturn of 2007 to 2009. In recent years, particularly from 2015 onwards, the HPI has seen an apparent increase, indicating a recovery and growth in the housing market.

Overall, the results indicate significant socioeconomic disparities among the counties analyzed. New Haven, Hartford, and Windham face persistent poverty challenges, while Fairfax and Fairfield show more robust economic indicators. Population growth is most pronounced in Fairfax, with Fairfield, Hartford, and New Haven also having large populations. Housing development patterns are particularly noteworthy in Fairfax and Fairfield counties. Household income and house price index trends further highlight the economic divergence across the counties.

7.1.4 Relationship of Features and Sale Price

In this section, we will examine the relationship of various features with our target variable, the house price. The relationships are analyzed using statistical patterns and observations. We binned most continuous features and then calculated each bin's mean and median house price. The resulting pattern was then examined. In our analysis, we only mention features for which a pattern regarding the development of the house price is observable. For the other features, there are no clear patterns.

7.1.4.1 Geographic Frequency Features

Most geographic frequency features do not exhibit a clear pattern regarding their effect on the sale price. However, some notable trends include the number of utilities, where higher frequency is associated with lower sale prices. The number of religious institutions, entertainment, and healthcare facilities show a quadratic relationship, with an initial decrease in sale prices followed by an increase (see Figure 9).

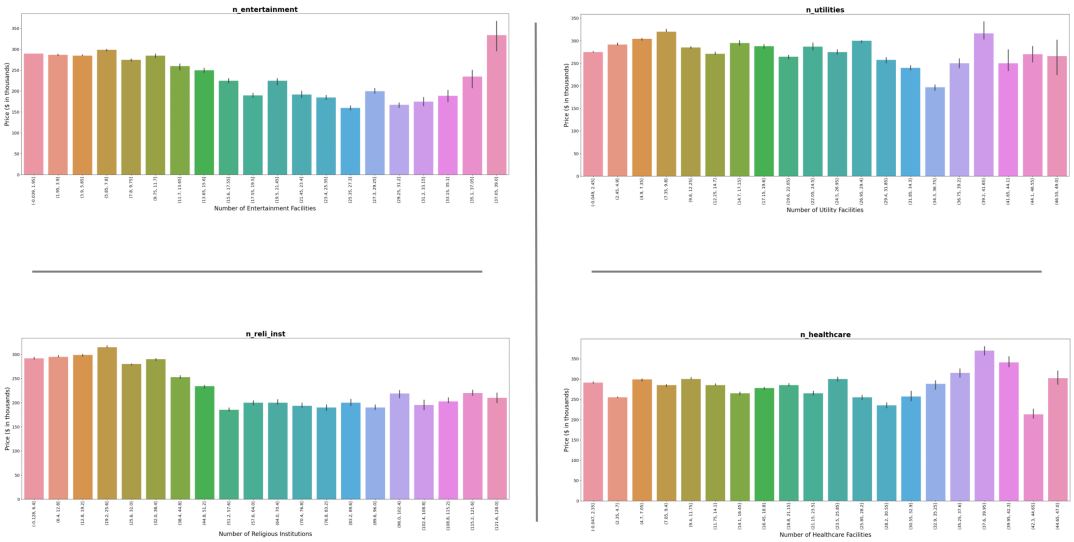


Figure 9: Geo-Frequency Features

7.1.4.2 Geographic Distance Features

The distance to various amenities reveals interesting and more apparent patterns than the frequency features. We observe a similar pattern for the distance to railway stations and museums: sale prices decrease with increasing distance. Although sale prices increase again at around 62 km from railway stations, this is

likely due to the smaller number of houses at greater distances affecting the mean price (see Appendix B).
Similar patterns emerge for the distance to hospitals and hotels. Sale prices increase slightly with increasing distance, followed by a sudden clear decrease (see Figure 10).

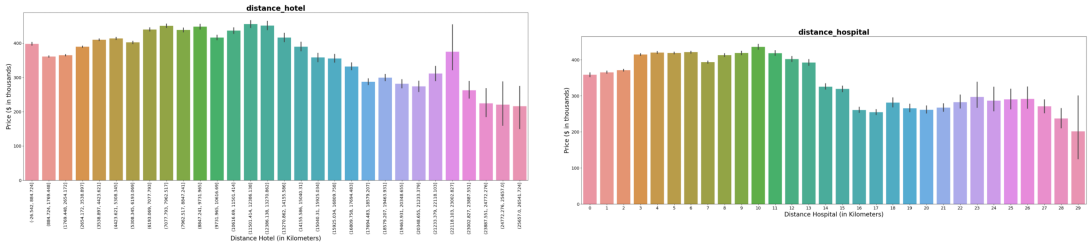


Figure 10: Distance to Hospital and Hotel

Sale prices increase slightly with increasing distance to airports, possibly because airports are typically located outside city centers (see Appendix B). Therefore, being near an airport could indicate being far from the city center. However, after a certain distance, sale prices decrease again, likely reflecting houses that are both outside the city center and far from the airport.

7.1.4.3 House Features

House features such as the number of bathrooms and bedrooms show distinct trends. Sale prices increase with the number of bathrooms up to four. It must be noted that the number of bathrooms varies greatly. However, most values are between zero and four. If we don't consider the outlier values (bigger than six) to avoid a biased picture through extreme cases, we can say that there is generally an increase in sale price with the increasing number of bathrooms (see Appendix B). The same picture can be seen with the number of half bathrooms (see Appendix B). The only difference is that the number of half bathrooms is overall lower. Most values are between zero and two. For the number of bedrooms, there is an increase in the sales price with an increasing number of bedrooms up until a number of five bedrooms. Beyond this number, the effect diminishes, likely due to fewer houses with more than five bedrooms. A similar pattern is observed in the living area, where prices increase with larger living spaces. However, there are instances where this trend does not always hold true (see Appendix B). The condition of the house is also a significant factor affecting sale prices. The number of houses in each condition category varies significantly, which must be considered since this affects the calculated mean prices. In particular, the number of houses in poor condition is very low, with a total number of 1537 compared to the

number of houses in average condition (total number of 610260). Therefore, we mainly analyzed the median price for this feature to decrease the effect of outliers. The median price is the lowest for houses in poor condition, while those in good condition have the highest. Overall, we get the expected result that the median price increases with better conditions (see Appendix B).

7.1.4.4 Sales Period

The period of sale also influences the sale prices. Mean sale prices increase slightly during the summer months (June, July, August) and decrease during winter (October, November, January). Over the years, mean sale prices increased until 1990, then decreased slightly, with a significant drop in 2008 and 2009, followed by a slight increase and another decrease in 2015 and 2016 (see Figure 11).

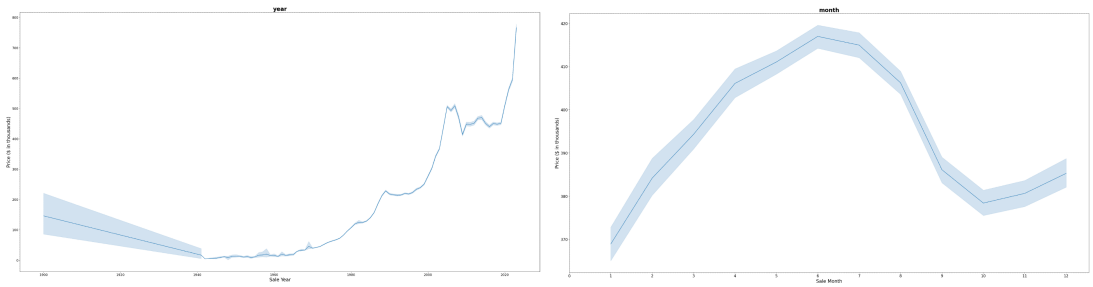


Figure 11: Sale Month and Sale Year

7.1.4.5 City and Age

Regarding the location of the houses, cities such as Greenwich, Westport, and New Canaan have the highest mean sale prices, with Greenwich being the highest at $2,072,040.71. Plainville, Norwich, and East Hartford have the lowest, with Plainville being the lowest with a mean price of $133,000. Regarding the age of the house, newer houses (0-20 years) and very old houses (155 years and older) have higher prices. The mean price decreases with age and then increases again (see Appendix B). Effective age shows similar trends, with mean prices decreasing until a certain point, after which they slightly increase. However, we have to keep in mind that older houses are less frequent in our dataset, resulting in individual transactions having a higher effect on the mean for these age ranges.
In conclusion, this analysis reveals varying influence patterns from different features on the sale price.

7.1.5 Geografic Analysis

This analysis aims to explain the relationship between house prices and various geographic features. We gain a clearer spatial understanding by plotting feature values and sale prices on maps based on house locations and aggregating these values within hexagonal grids. This method allows us to calculate the mean values for each hexagon based on the houses located within them, thereby facilitating a detailed geographic analysis. In Figure 12, we have plotted the mean price for different hexagonal areas. The closer the hexagon's color is to yellow, the higher the price. The same applies to other figures, such as Figure 13. The only difference is that we plot the distance to the airport instead of the price. A greater distance is again represented by the color yellow.

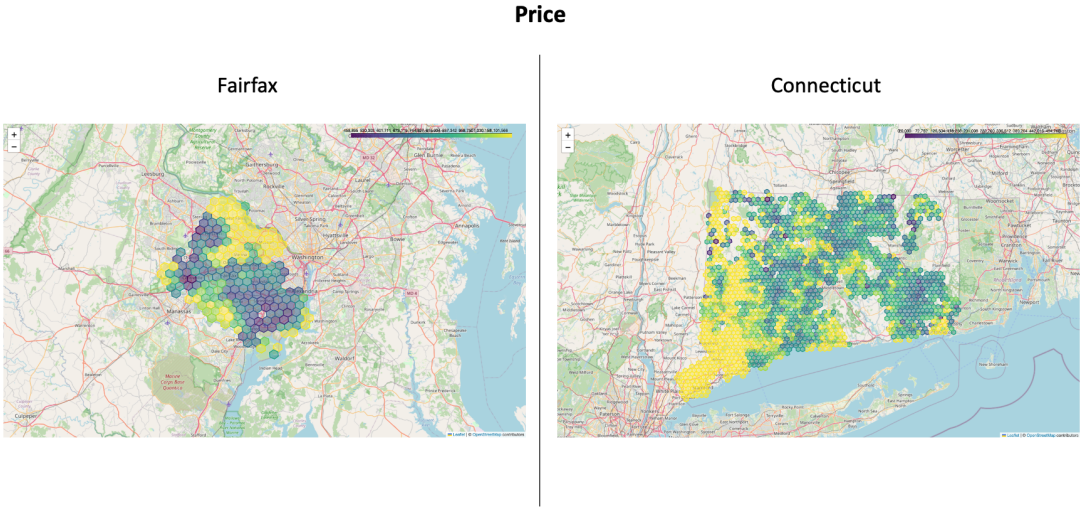


Figure 12: Price

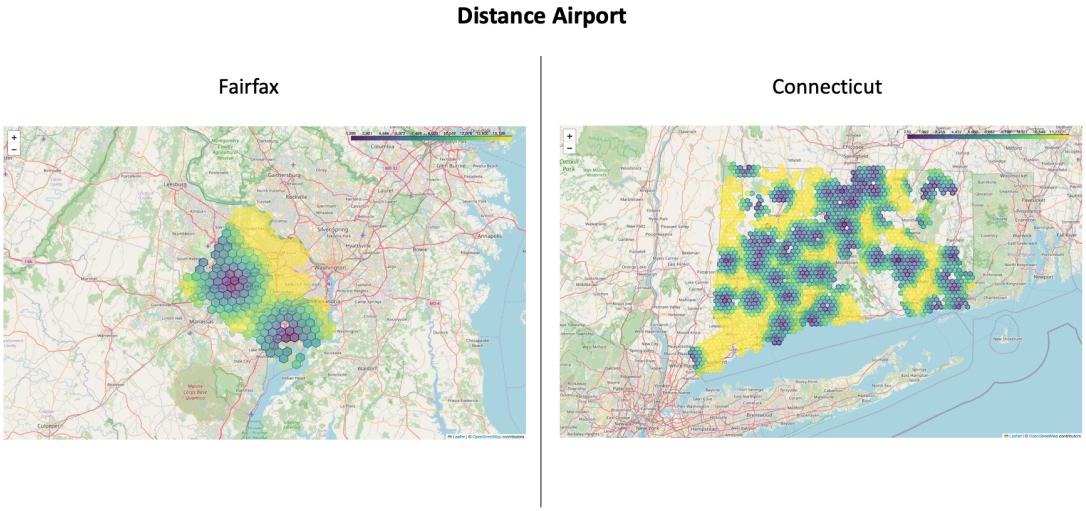


Figure 13: Distances to Airports

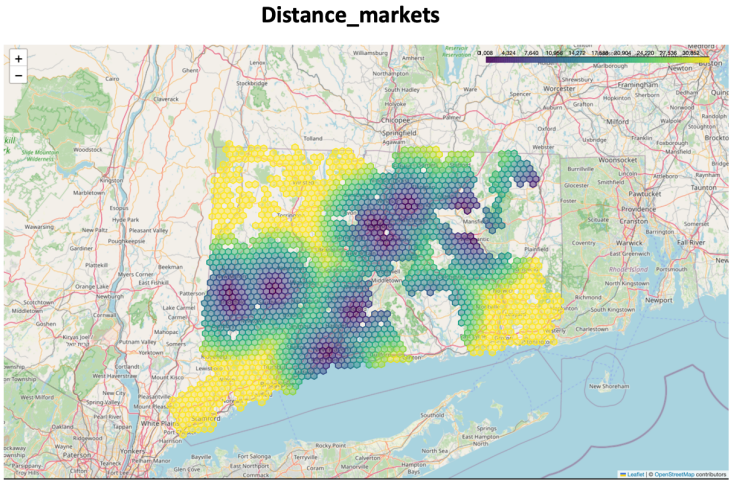


Figure 14: Distances to Markets in Connecticut

Upon plotting the distance to markets, a noticeable pattern emerges. Areas with larger distances to markets tend to have higher sale prices, a trend also reflected by a relatively high correlation coefficient (see Figure 14). This is particularly evident in Fairfield County, where houses far from markets exhibit higher sale prices. Similar patterns are observed in the southeastern part of New London and the northwestern part of Litchfield, demonstrating the correlation between greater distances to markets and increased sale prices in these regions.

A similar trend is observed with the distance to airports (see Figure 13). Areas with larger distances from airports often have higher sale prices. This pattern is particularly evident in Fairfield County, where houses are typically far from airports. In New Haven, southern Litchfield, and Hartford, lower distances to airports correspond with lower sale prices. Notably, the southeastern edge of New

Haven County, which has the largest distances to airports within the county, also records the highest sale prices. This relationship can also be observed in Fairfax County, thereby underlining the generality of this pattern.

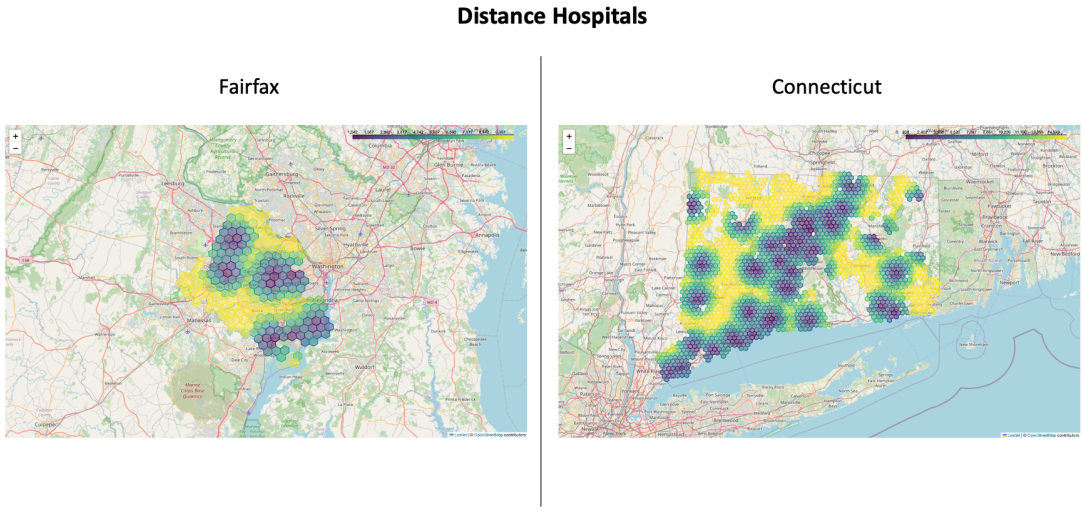


Figure 15: Distances to Hospitals

The relationship between distance to hospitals and sale prices is less straightforward. In Fairfield County, properties closer to hospitals tend to have higher sale prices. However, in other areas such as western Litchfield, southeastern New Haven, and southeastern New London, properties with moderate distances to hospitals (neither too close nor too far) show higher sale prices (see Figure 15). The correlation coefficient does not clearly capture this nuanced relationship, indicating a non-linear relationship between sale prices and the distance to hospitals.

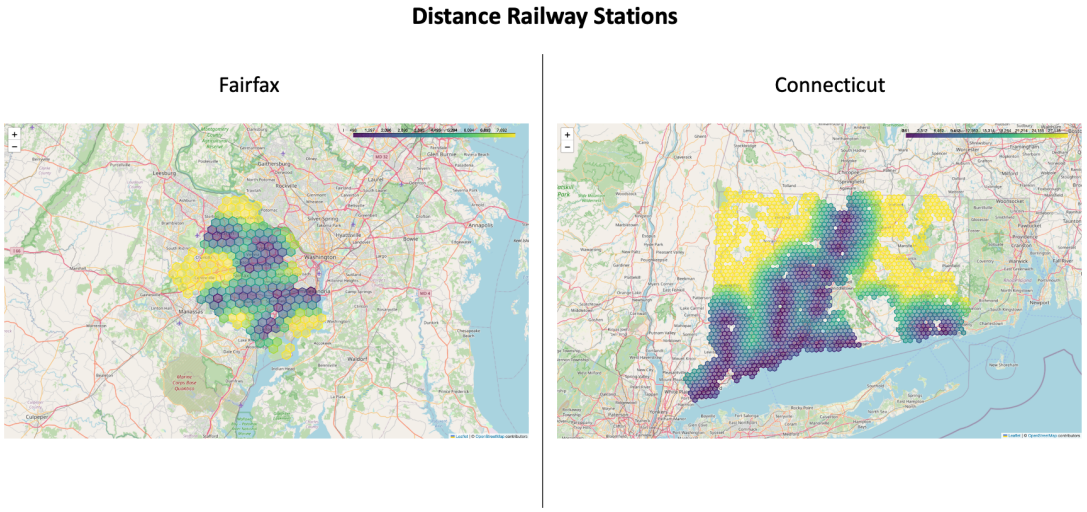


Figure 16: Distances to Railway Stations

The analysis of railway station distance reveals that areas with shorter distances to railway stations generally have higher sale prices. This trend is particularly

noticeable in Fairfield County, New London, New Haven, and Fairfax County. However, this relationship is not consistently observed in Hartford and Tolland counties, suggesting regional variations in the importance of railway station proximity to housing prices (see Figure 16).
In Fairfax County, the number of emergency facilities shows a negative relationship with sale prices, indicating that higher numbers of these facilities correlate with lower housing prices (see Figure 17).
Fairfax County also exhibits a negative correlation between the number of food and drink establishments and sale prices. Additionally, a similar negative relationship is observed with the number of religious institutions and sports facilities, suggesting that higher densities of these amenities may correspond to lower housing prices in this area (see Figure 17). However, these relationships can only be seen in Fairfax County and are not observable in Connecticut.

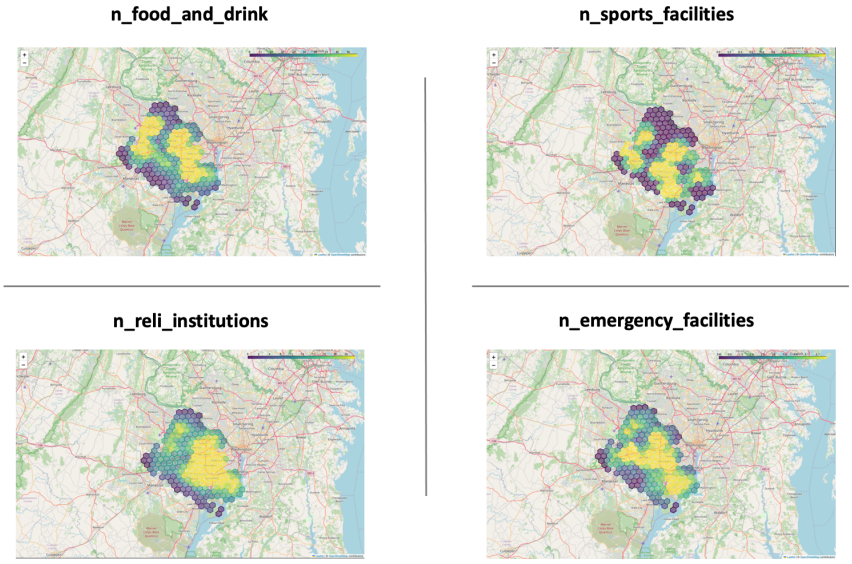


Figure 17: Geo-Frequency Features Fairfax County

This geographic analysis highlights the complex and varied relationships between housing sale prices and proximity to various features. While some trends, such as the positive correlation between distance to markets and airports with sale prices, are clear, other relationships, such as those involving hospitals, exhibit more complexity. These findings underscore the importance of considering regional variations and the specific context when analyzing real estate markets.

7.2 Data Analysis Appreciation Prediction

When analyzing the dataset for our appreciation prediction, we found that the correlation pattern between the features is the same as for the dataset used for

the initial house price prediction model. Additionally, features such as appreciation_time, prior_year, prior_month, and prior_price do not have any remarkable correlations with other features. Overall, no feature shows a high correlation with appreciation. The feature with the highest correlation is appreciation_time, which only has a correlation of 0.07. Following that is prior_year with a negative correlation of -0.06 and prior_price with -0.04 (see Figure 18). No obvious or clear patterns are observed when we plot the different features against our target.

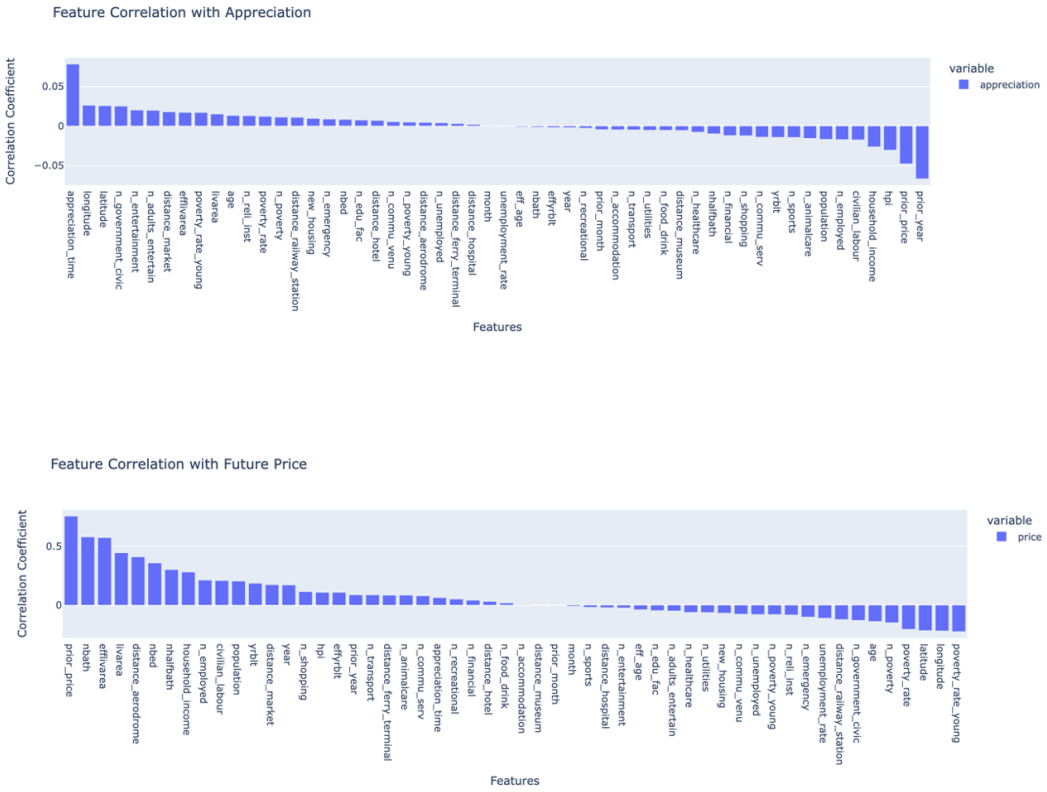


Figure 18: Correlation with Different Target Variables

This indicates that it might be better to indirectly predict appreciation by predicting the future price and then manually calculating the appreciation. Looking at the correlations with the future price, we can observe the same correlation pattern as for the initial price. The main difference is that prior_price, as expected, has the highest correlation with the future price at 0.76 (see Figure 18). This supports the idea of predicting the future price and then calculating the resulting predicted appreciation instead of directly predicting the appreciation.

7.3 Modelling

As stated in the business model description, our framework requires the development of two machine learning models. The first model aims to predict the current

house price, accurately reflecting the market value by considering the prevailing market and economic conditions. This model's precision is at the core of the business model, as the subsequent calculation of value appreciation is based on these predictions. It is essential to obtain accurate house prices to ensure fairness. An underprediction of the initial price, followed by an overestimation of the house price later on, would result in an inflated appreciation and a higher profit for the borrower to pay the investor. Conversely, overestimating the initial price would result in lower appreciation and reduced profit for the investor.
In this section, we iteratively and successively develop the optimal machine learning model and evaluate its performance at each step. However, before beginning the actual modeling, we must preprocess the data, building upon the processed data as described in the data section.

7.3.1 Modelling Specific Data Preparation

In the initial step of our data preparation, we removed data points with faulty data. Data points with a negative age were excluded. For data points with a negative effective age, the negative value was replaced with the normal age. Additionally, data points with a construction year or effective construction year of zero, which would result in excessively large house ages, were also removed. Missing values for the effective age and effective build year were substituted with the normal age and build year, respectively. Missing values for the effective living area were replaced with the normal living area.
We encountered missing values for economic factors due to the availability of economic data only from 1990 onwards, while house sales data included transactions before 1990. These missing economic factors for earlier transactions were replaced with those from 1990. This substitution is legitimate as our model does not aim to predict future sale prices or appreciations, making future developments irrelevant to the prediction task. In real-world applications, it is reasonable to assume that the most current economic factors are accessible.
Since certain models, like Extreme Gradient Boosting (XGBoost) – an advanced and efficient implementation of gradient boosting – cannot process categorical variables, we encoded the categorical variables in our dataset. These features included county, state, condition description, and city. With nine counties, two states, and five house conditions, one-hot encoding was feasible without causing an extreme increase in dimensionality in our dataset. One-hot encoding converts categorical data into a binary format, where a unique binary vector represents each category. This process adds a column for each category, which indicates the presence of a particular category through a binary value.
However, for the city feature, which includes 149 unique cities, one-hot encoding

would lead to an explosion in dimensionality. Consequently, we employed ordinal encoding for the city feature, assigning a unique integer to each city. While this creates a ranking or order of the cities, which is problematic for linear regression models that assume meaningful ordinality, it is less of a concern for tree-based models, which can handle categorical variables effectively.
For data splitting, we implemented a random train-test-validation split. The training set was used to train our machine learning models, the validation set for fine-tuning through hyperparameter tuning, model architecture and feature selection decisions. The test set was reserved for evaluating the final model performance without influencing any decisions based on this dataset.
Since the dataset includes multiple transactions of the same house, we based our split on the house ID to ensure that each house is only contained in one dataset and to evaluate the model performance on transactions of houses not used in the training process. We randomly assigned 70% of the houses to the training set, 15% to the test set, and 15% to the validation set. After preprocessing and splitting, the training set included 701,684 transactions, the test set 149,892 transactions, and the validation set 150,432 transactions.

7.3.2 Baseline Model

After preprocessing and splitting the data, we can now focus on the modeling phase, which involves developing the most effective machine learning models for our house price prediction task.
In the initial step, we fit an XGBoost model using all the features and apply hyperparameter tuning to determine the optimal hyperparameters that yield the best performance on the validation set. This model's performance will serve as our baseline for further architectural and modeling decisions.
For the XGBoost model with the following hyperparameters:

Parameter	**Value**
Objective Function	squared error
Number of Estimators	350
Learning Rate	0.1
Max Depth of Trees	10

Table 6: Hyperparameters for the XGBoost Model

We observed the following model performance:

Metric	Value
MSE	39,580,179,840.86
RMSE	198,947.68
MAE	59,818.08
MAPE	27.78
SMAPE	15.64
R^2	0.810732

Table 7: Baseline Model Evaluation Metrics

The model demonstrates a good fit with an R^2 value of 0.81, indicating that it explains a significant portion of the variance in house prices. The error metrics (MSE and RMSE) suggest high prediction errors, likely due to the presence of significant outliers that disproportionately affect the RMSE and MSE. This is underscored by the RMSE being substantially higher than the MAE.
The percentage error metrics (MAPE and SMAPE) show that the model has a reasonable average percentage error, with a MAPE of 27.78% and a SMAPE of 15.64%.

7.3.3 Comparing Different Models

After having a baseline performance, we want to evaluate other tree-based model architectures. Specifically, we tested a decision tree, random forest regressor, and gradient-boosting regressor.
We observed the following validation set performance for the different models:

Metric	Decision Tree	Random Forest	GBR	XGB
MSE	67,654,896,543.76	41,630,437,816.74	**39,060,591,569.46**	39,580,179,840.86
RMSE	260,105.55	204,035.38	**197,637.53**	198,947.68
MAE	96,336.51	61,305.39	60,480.64	**59,818.08**
MAPE	**24.83**	27.39	26.36	27.78
SMAPE	25.16	**15.24**	15.92	15.64
R^2	0.676483	0.800929	**0.813217**	0.8107326

Table 8: Validation Set Performance Metrics for Different Models

When compared to the different models, the Decision Tree Regressor has the highest error metrics and the lowest R^2 value, indicating the poorest performance. It struggles significantly with both absolute and percentage errors. The Random Forest Regressor performs better than the Decision Tree but still has higher error metrics than the boosting methods. It performs relatively well in percentage errors, but overall, it is less favorable. The Gradient Boosting Regressor stands out with the lowest MSE and RMSE and the highest R^2 value, indicating it explains the most variance in house prices. It makes the least absolute percentage errors on average and performs well across most metrics. The XGBRegressor

shows strong performance with the lowest MAE but struggles with percentage errors, making it a close second to the Gradient Boosting Regressor (see Table 8). Despite the Gradient Boosting Regressor demonstrating the best performance, we continued using the XGBoost Regressor. This decision was based on XGBoost's significantly higher computational efficiency and its performance, which is reasonably close to that of the Gradient Boosting Regressor.

7.3.4 Effect of Different Feature Categories

The features used for our house price prediction model can be categorized into four distinct groups. The first category are the core house features, which directly describe the properties of the house, such as age, living area, and the number of bedrooms. The second category includes geographic distance features, which indicate the proximity of the house to various amenities, such as the nearest airport or hospital. The third category consists of geographic frequency features, such as the number of restaurants, representing the count of different types of amenities within a 5 km radius of the house. The final category encompasses economic features, such as the unemployment rate, reflecting the respective counties' economic conditions at the time of sale.
In the subsequent step, we aim to evaluate the impact of these different feature categories on the model's performance. We will begin with the core house features as our baseline and then incrementally add the other feature categories to observe their effects on model performance.

Features	MSE	MAE	MAPE	R^2	RMSE	SMAPE
Only Core Features	42,381,991,448.51	61,929.09	29.21	0.797335	205,868.87	15.88
With Geo Distance Features	**39,152,079,796.55**	**59,735.27**	**26.47**	**0.812780**	**197,868.84**	**15.59**
With Geo Frequency Features	40,173,898,409.03	60,655.72	27.84	0.807894	200,434.27	15.71
With Economic Features	41,354,600,075.69	62,432.27	30.43	0.802248	203,358.30	16.08
With All Geo Features	39,265,351,898.72	59,794.53	27.50	0.812238	198,154.87	15.61
With All Features	39,580,179,840.86	59,818.08	27.78	0.810733	198,947.68	15.64

Table 9: Performance Metrics for Different Feature Sets

Starting with only core features already yields a reasonably high R^2 value of 0.797. When geo distance features are added, the model's performance significantly improves, with the MSE dropping significantly and the R^2 value increasing to 0.8128, indicating that these features enhance the model's accuracy the most (see Table 9).
Adding geo-frequency features also improves the model compared to core features alone, but not as significantly as geo-distance features, as seen in the R^2 value of 0.8079. Including economic features also improves performance compared to using only core features. However, the improvement is less than when adding the geo features. This could be due to the fact that we only have the economic

factors at the county level and not at the city level.
Combining all geo features results in a similar performance to using geo-distance features alone, with an R^2 value of 0.8122. Using all features doesn't significantly improve the model's performance beyond using geo-distance features, with an R^2 value of 0.8107.
Overall, incorporating geo-distance features provides the best performance improvement, followed by geo-frequency features and economic features, all enhancing the model's accuracy compared to using only core features. Since adding all features together does not seem to be the best approach, this indicates the need for a smart feature selection strategy. It also suggests that the ability of tree-based models to select the best features themselves does not always lead to the best overall performance.

7.3.5 Feature Selection

For our feature selection, we used Recursive Feature Elimination (RFE). This method recursively considers smaller and smaller sets of features. Using our XGBoost model and the feature importance obtained from it, we start with all our features and train the model. We then use the trained model and the resulting feature importance to prune the least important features from the feature set.
The results show that the following set of 50 features leads to the best achievable performance according to the applied RFE technique:

1. **Core Features**: city, yrblt, effyrblt, nbed, nbath, nhalfbath, livarea, efflivarea, year, age, eff_age, longitude, latitude, county_Fairfield, county_Litchfield, cond_desc_Average, cond_desc_Fair, cond_desc_Good, cond_desc_Poor
2. **Geo Distance Features**: distance_aerodrome, distance_ferry_terminal, distance_railway_station, distance_market, distance_hospital, distance_hotel, distance_museum
3. **Geo Frequency Features**: n_reli_inst, n_edu_fac, n_healthcare, n_emergency, n_animalcare, n_commu_venu, n_commu_serv, n_food_drink, n_financial, n_transport, n_entertainment, n_sports, n_utilities, n_accommodation, n_government_civic, n_recreational
4. **Economic Features**: hpi, household_income, new_housing, population, n_poverty, n_poverty_young, unemployment_rate, n_employed,

We observed the following performance:

Metric	After Feature Selection	All Features
MSE	**39,127,435,488.81**	39,580,179,840.86
MAE	60,071.33	**59,818.08**
MAPE	28.02	**27.78**
R^2	**0.812898**	0.810733
RMSE	**197,806.56**	198,947.68
SMAPE	15.69	**15.64**

Table 10: Performance Metrics Before and After Feature Selection

Feature selection has led to an improvement in several metrics, including MSE, R^2, and RMSE, suggesting a marginal enhancement in the model's overall predictive performance. However, there is a minor increase in some error metrics (MAE, MAPE, and SMAPE), indicating a trade-off. Overall, the impact of feature selection can be seen as positive, especially since the target cost function being minimized is the mean squared error, which also optimizes R^2. To achieve significant improvements in other metrics, it would be necessary to use a different cost function for the optimization.

7.3.6 Feature Extraction PCA

After selecting a subset of features that led to the best model performance, we try to extract new features from this subset. One method for extracting new features is PCA, which generates features as linear combinations of the original features.
Upon experimenting with different numbers of principal components (PC), we observed the best validation set performance when adding three principal components to our feature set.
After incorporating these principal components into our features, the results are as follows:

Metric	After Adding PC	Before Adding PC
MSE	**38,594,474,450.04**	39,127,435,488.81
MAE	60,113.77	**60,071.33**
MAPE	28.10	**28.02**
R^2	**0.815446**	0.812898
RMSE	**196,454.76**	197,806.56
SMAPE	15.71	**15.69**

Table 11: Model Performance Before and After Adding Principal Components.

Adding principal components to the features resulted in a noticeable decrease in the Mean Squared Error (MSE) and Root Mean Squared Error (RMSE), indicating better model accuracy. The R^2 value also increased, suggesting a higher

proportion of variance explained by the model. Although the MAE, MAPE, and SMAPE increased slightly, these changes are minor and can be overlooked. Overall, the principal components enhance the model's accuracy and explanatory power, as evidenced by the improvements in MSE, RMSE, and R^2.

7.3.7 Feature Extraction Clustering

Building upon that, another method that can be used to generate new features is clustering. We use k-means clustering to add the resulting clusters as additional features. After trying out and evaluating different numbers of clusters and their effects on the model's performance, we observed the best model performance with six clusters.

We found that simply adding one feature that assigns each data point to a cluster does not improve the model's performance; rather, it decreases it. However, adding six features, where each feature describes the distance of each data point to a specific cluster centroid, leads to an improvement in the model's performance. This approach resulted in the following performance metrics:

Metric	After Adding Cluster Distances	Before Adding Cluster Distances
MSE	**37,854,193,451.47**	38,594,474,450.04
MAE	**59,940.46**	60,113.77
MAPE	28.9313	**28.0982**
R^2	**0.818986**	0.815446
RMSE	**194,561.54**	196,454.76
SMAPE	**15.7090**	15.7142

Table 12: Performance Metrics After Clustering

Adding cluster distances as additional features resulted in several improvements in model performance. The MSE and RMSE decreased, indicating enhanced model accuracy. The R^2 value increased, suggesting a higher proportion of variance explained by the model. The MAE also decreased slightly, reflecting a minor improvement in the average absolute difference between predicted and actual values. The SMAPE remained nearly the same. However, the MAPE increased slightly, which can be considered negligible. Overall, adding cluster distances improved the model's accuracy and explanatory power.

7.3.8 Hybrid Model Architecture

To further enhance the model's performance, we implemented a hybrid model architecture inspired by Özöğür-Akyüz et al. (2022). The core idea of this approach is to fit different machine learning models for distinct clusters within the initial dataset.

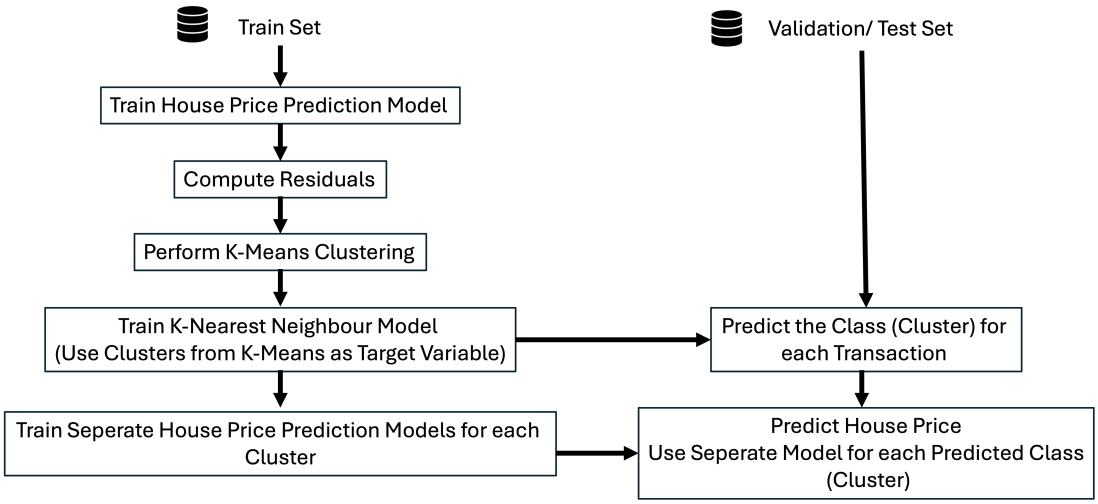


Figure 19: Hybrid Model Architecture

The first step involves fitting a regular machine learning model to the entire training set. Next, we calculate the residuals by determining the difference between the predicted house prices and the actual house prices. We then apply k-means clustering to the training dataset using these residuals (see Figure 19).
The cluster allocation from this step is used as a classification target. We fit a classification model, such as k-nearest neighbors, to the training dataset with the cluster allocation as the target variable. This trained model is then used to classify our validation set.
Subsequently, we split the training set based on the different clusters. For each cluster, we fit a separate model with distinct hyperparameters. The validation set is also divided based on the predicted classes and used to validate each of the different machine learning models (see Figure 19).
In our case, we slightly adjusted this model architecture. Instead of basing the calculation of the different clusters solely on the residuals, we included all other features, which led to better performance overall. We found that working with four clusters yielded the best results, and therefore, we fit four different models accordingly.

Metric	Cluster 0	Cluster 1	Cluster 2	Cluster 3	Final Performance	Before Hybrid Architecture
MSE	64,721,184,222.014	5,842,396,543.76	42,761,942,843.67	122,760,968,717.73	**36,767,979,258.64**	37,854,193,451.47
MAE	79,907.50	32,623.35	73,522.78	100,444.18	**57,548.84**	59,940.46
MAPE	49.68	0.07684	23.94	179.05	**28.57**	28.93
R^2	0.802018	0.950044	0.542442	0.695064	**0.824180**	0.818986
RMSE	254,403.59	76,435.57	206,789.61	350,372.61	**191,749.78**	194,561.54
SMAPE	19.97	7.36	25.53	30.55	**14.93**	15.71

Table 13: Model Performance Metrics for Different Clusters and Overall Performance

Cluster 1 showed the best performance with the lowest errors (MSE, MAE, MAPE, RMSE, SMAPE) and the highest R^2, indicating excellent predictive accuracy and stability (see Table 13). Cluster 3 had the worst performance, with the

highest errors and the lowest stability, as reflected by the highest MAE, MAPE, and a lower R^2 value. Cluster 0 and Cluster 2 had moderate performance, with Cluster 2 struggling to explain variance (lowest R^2) but having lower errors than Cluster 0. The overall model performance improved when combining the performance of the different clusters, with lower MSE, RMSE, and a high R^2 value, indicating good accuracy and variance explanation. Especially if we compare the overall performance with the performance before applying the hybrid model architecture, we can see a significant decrease in the MSE and the MAE. In total all metrics improved through this hybrid architecture.

7.3.9 Final Model Test Set Performance

Throughout the previous steps, we consistently evaluated our model on the validation set. This evaluation process guided our architectural decisions and hyperparameter tuning. To ensure a comprehensive assessment, we retrained our model on the combined train and validation data. Subsequently, we conducted a final evaluation using the previously untouched test data set.
For our final hybrid model architecture, we achieved the following performance metrics on the test set:

Metric	**Value**
MSE	39,454,083,894.08
RMSE	198,630.52
MAE	60,759.09
MAPE	22.80
SMAPE	16.03
R^2	0.812289

Table 14: Baseline Model Evaluation Metrics

The final model does not perform as well on the test set compared to the validation set. This is expected, as we performed hyperparameter tuning and made architectural decisions based on the validation set performance, indirectly optimizing the model for it. However, the model still demonstrates reasonably good performance on the test set. Notably, the R^2 value of 0.8123 indicates that the model captures a significant amount of the house price variance on unseen data. Additionally, the MAPE is 22.08, which is lower than that of the validation set. Overall, the R^2, MSE, RMSE, and MAPE on the test set are all better than the baseline model's performance on the validation set.
In summary, our final model architecture for house price prediction achieves reasonably high performance on unseen data.

7.3.10 Feature Importance

In the following section, we will analyze the feature importance of our XGBoost model (prior to applying the hybrid model architecture) to better understand which features significantly contribute to improved predictions. We employed three methods to determine feature importance: permutation importance, SHAP values, and XGBoost feature importance. This multi-method approach helps mitigate the weaknesses of individual methods, providing a more holistic view of feature importance.

Permutation importance measures the change in the model's performance when a feature's values are shuffled. A larger decrease in performance indicates a more important feature. SHAP values provide insight into each feature's impact on the model's predictions, with higher SHAP values indicating a greater contribution. The SHAP summary plot also shows the direction of the feature's impact (positive or negative). XGBoost's feature importance is based on the information gain through splits involving the feature across all trees in the model.

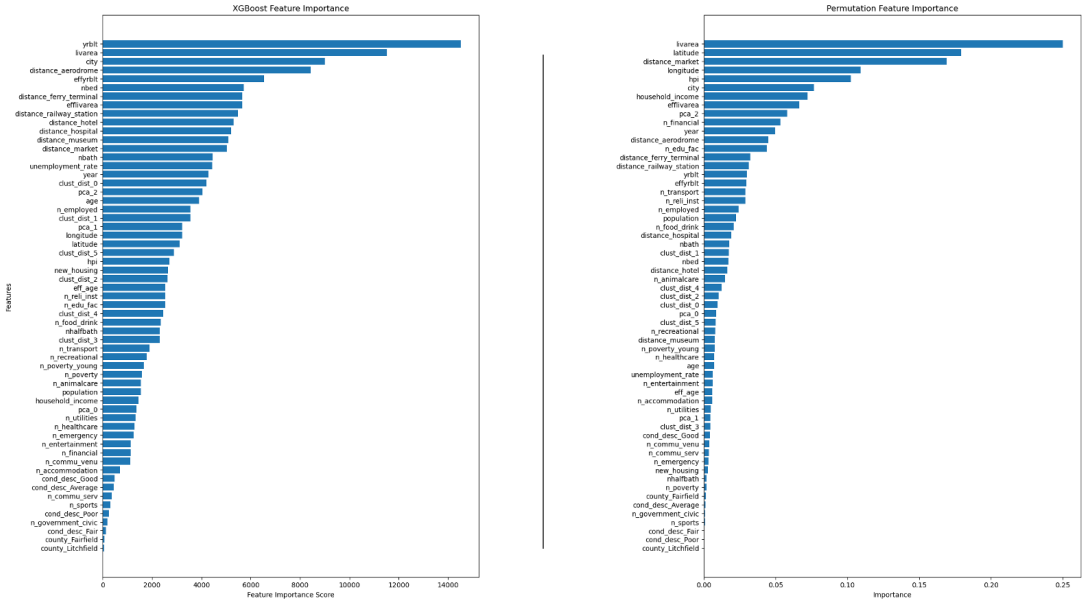


Figure 20: XGB and Permutation Importance

The key findings include:

1. **Living Area**: The living area consistently ranks as one of the top two most important features across all three methods. The additional inclusion of the effective living area in the top 10 most important features generally underscores its crucial role in predicting house prices.
2. **Economic Factors**: Both household income and HPI are consistently among the most important features across all three methods.

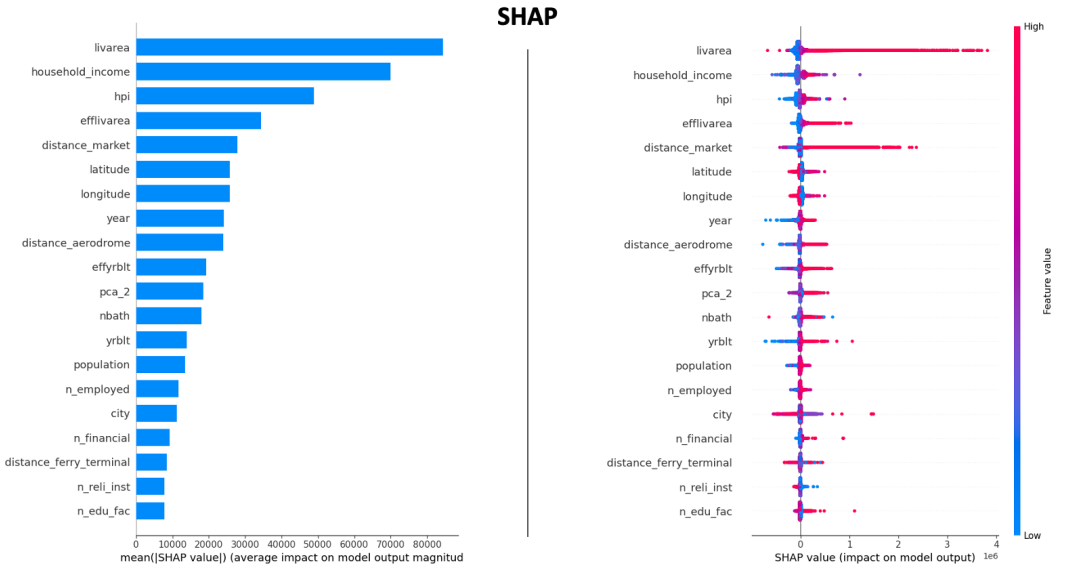


Figure 21: SHAP Importance

3. **Geographic Features**: Regarding the geographic features, mostly distance features are among the most important, whereas this is not the case for the frequency features. Only permutation importance uniquely identifies the number of financial institutions as a top feature. According to permutation importance and SHAP values, the distance to the market is the most important geographic feature. In contrast, XGBoost feature importance highlights the distance to the airport, nearest ferry terminal, nearest railway station, and nearest hotel as top features.
4. **Latitude and Longitude**: These geographic features are consistently among the most important across all three methods.

Overall, this analysis underlines the significant role of living area, household income, HPI, and various geographic distance features in predicting house prices, providing a comprehensive understanding of feature importance in our XGBoost model.

7.3.11 House Price Appreciation Prediction

As previously described, the second required machine learning model predicts the appreciation of house prices in the upcoming years. This prediction model is particularly relevant for investors, as it informs their investment decisions by identifying houses with the highest predicted appreciation.
In a real-world scenario, we will not have the actual house price (sale price) at the time of investment, so we must rely on the predicted house price. Consequently, our target variable for appreciation prediction will be the appreciation of the

predicted house price.
The prediction process will be as follows:

1. First, we predict the current house price and use this as an additional feature in the model that predicts the future house price.
2. With the predicted initial house price and the predicted future house price, we calculate the predicted appreciation.
3. Finally, we evaluate the performance of this architecture by comparing the predicted appreciation with the appreciation calculated using the predicted initial house price and the future sale price of the house (true future house price).

$$appreciation = \frac{future\_house\_price}{predicted\_initial\_house\_price}$$

$$\widehat{appreciation} = \frac{\widehat{future\_house\_price}}{predicted\_initial\_house\_price}$$

7.3.12 Data Preprocessing

Regarding the data preprocessing steps, we will follow the same procedures as for the house prediction model, with one key difference: we will exclude any transactions prior to 1990. In the preprocessing steps for the house price prediction model, we replaced missing values for economic factors before 1990 with data from 1990, effectively using future values. This approach is problematic for our model, which aims to predict future house prices, as it would involve using future information to predict future target values.

7.3.13 Model Performance

In the following section, we will experiment with different model configurations to predict future house prices and, consequently, their appreciation. We will utilize the same configurations that were applied to our house price prediction model.
Before predicting future house prices, we will validate our approach by evaluating the model's performance when directly predicting appreciation, incorporating the predicted initial house price as a feature. The model's performance results are as follows:

Metric	Value
MSE	10.5203
MAE	0.2698
MAPE	2.4744
R^2	-0.7895
RMSE	3.2435
SMAPE	58.5393
RMSPE	73.0521

Table 15: Model Performance Metrics

As anticipated from our data analysis, where we found that there are no clearly correlated features with the target, nor observable patterns between features and the target, we observe poor performance when attempting to directly predict appreciation. This outcome supports our approach of first predicting the future house price and then manually calculating appreciation.
For the model performance when predicting future house prices with different model architectures and feature configurations, we observe the following results:

Metrics	Initial Price Prediction	Future Price Prediction	Appreciation Prediction
All Features			
MSE	21,704,888,896.83	29,654,861,197.96	**0.5374**
MAE	58,299.64	**64,236.57**	**0.2093**
MAPE	0.3147	**0.1741**	**2.2180**
R^2	0.825888	0.848297	0.908593
RMSE	147,325.79	172,205.87	**0.7331**
SMAPE	17.92	**13.17**	57.47
After Feature Selection			
MSE	21,746,812,144.42	29,284,990,823.64	3.8789
MAE	58,951.14	64,543.59	0.2216
MAPE	0.3118	0.1744	2.2620
R^2	0.825552	0.850190	0.821916
RMSE	147,468.00	171,128.58	1.9695
SMAPE	18.18	13.29	57.94
After Feature Extraction			
MSE	21,857,016,793.08	**28,792,039,967.04**	0.9145
MAE	59,325.25	64,902.87	0.2132
MAPE	0.3177	0.1748	2.7336
R^2	0.824668	**0.852711**	**0.919279**
RMSE	147,841.19	**169,682.17**	0.9563
SMAPE	18.23	13.36	58.12
Hybrid Model			
MSE	**21,447,522,050.37**	41,403,052,151.22	0.7138
£ MAE	**57,950.50**	80,511.48	0.2115
MAPE	**0.3090**	0.2340	4.0943
R^2	**0.8280**	0.8502	0.8800
RMSE	**146,449.72**	203,477.40	0.8449
SMAPE	**17.89**	17.15	**57.13**

Table 16: Performance Metrics for Different Model Configurations

As expected, for the initial price prediction, the hybrid model architecture performed the best across all metrics. Interestingly, this is not the case for the future price prediction model. In this scenario, we observe the best performance from the

baseline model with all features and the model that includes extracted features through PCA and clustering. Both models exhibit very similar performance.
The model with extracted features demonstrates better performance for the squared error metrics (MSE, RMSE) and R^2. In contrast, the baseline model shows better performance for the absolute error metrics (MAE, MAPE) and SMAPE.
When examining the appreciation values and prediction performance, we see the highest R^2 value for the models with extracted features. However, the baseline model performs slightly better in other performance metrics.
In summary, we will use the hybrid model for the initial price prediction and the model with extracted features for future price prediction. Using these models, we will proceed to test and evaluate our business model.

7.3.14 Feature Importance

After evaluating the feature importance for the future house price prediction model, we can overall observe a similar set of the most important features compared to the initial house price prediction model. The most obvious difference is that the most important feature of the future house price prediction model is the predicted initial price. Additionally, the house price index and the sale year have notably increased in importance (see Figure 22).

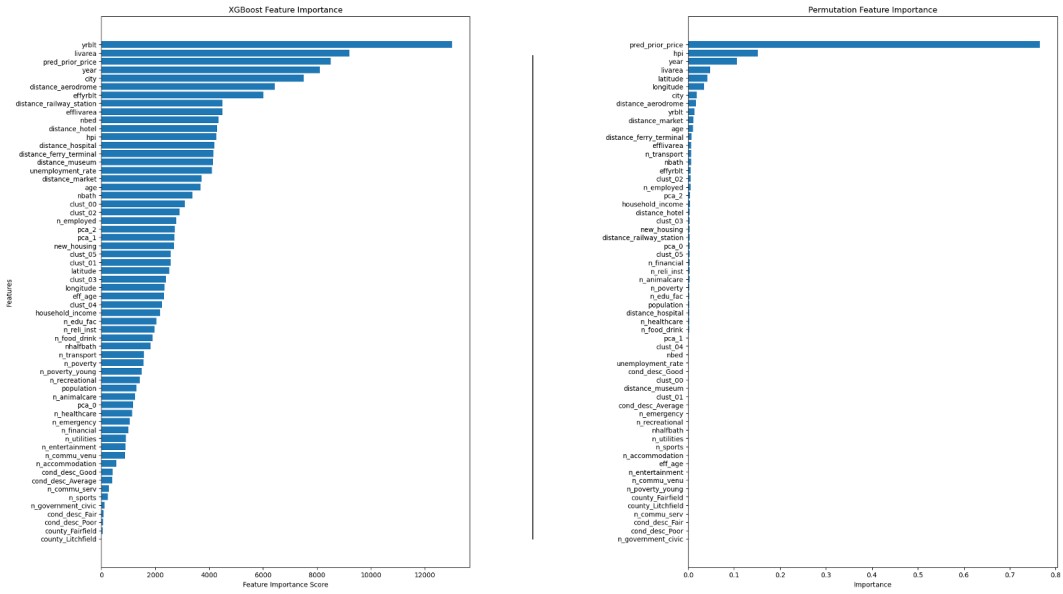


Figure 22: XGB and Permutation Importance

7.4 Evaluating the Business Model

7.4.1 Experiment Setup

To test our business model and the trained machine learning models in a real-world scenario, we have set up the following experiment. We will work with 1,000 homeowners who need funds and are willing to offer a percentage of their house's value appreciation over five years in exchange for a loan. An investor with a budget of $50,000 will provide these loans in return for a share of the houses' value appreciation. To mitigate risk, the investor will diversify by investing in multiple houses, limiting each investment to one percent of the initial house price in exchange for one percent of the value appreciation.
We considered houses purchased in 2017 and sold again in 2022, randomly selecting 1,000 such houses for the study. To train our house price prediction and future house price prediction models, we only used transaction data up to 2017. First, we built our initial house price prediction model using the proposed hybrid architecture. Next, we developed our future house price prediction model using the model with extracted features (via PCA and clustering).
After training, we used these models to predict the initial house prices in 2017 and the future house prices in 2022 for the selected 1,000 houses. Using the predicted prices, we calculated the value appreciations.
From the 1,000 houses, we selected those with the highest predicted value appreciations for investment, prioritizing until the $50,000 budget was exhausted. Each selected house received an investment of one percent of its predicted initial price, except for the last house, which received the remaining budget if it was less than one percent of the initial price.
Firstly, we will evaluate the business model addressing the following key points:

1. Evaluate the profit from the investment.
2. Compare the profit to a scenario where houses are selected randomly.
3. Assess the profit under a perfect selection scenario where the future house prices are known, but the initial prices are based on predictions, reflecting real-world constraints where future prices become known only upon sale.

7.4.2 Profit Calculation

To evaluate the profitability of our investment strategy, we followed a structured approach. First, for each selected house, we computed the absolute value appreciation as the difference between the true future house price and the predicted initial price since the true initial price would likely remain unknown in real-world scenarios. In cases where the appreciation was negative, we set these values to

zero, as the decrease in house value is not shared with the investor. In such cases, the investor only gets back the borrowed money without any profit share.
Next, we calculated the profit share by determining one percent of the absolute value appreciations. For the last selected house, where the invested sum did not equal one percent of the predicted initial price, we adjusted the appreciation by multiplying it with the corresponding fraction of the investment. We then summed all the profit shares from all investments to obtain the total profit.

7.4.3 Comparison of Different Selection Strategies

We compared three different selection strategies. In the random selection strategy, houses were chosen randomly without considering predicted appreciations. In the perfect selection strategy, we based our choices on the value appreciation calculated using the predicted initial price and the true future price, even though the future price would not be known at the time of the investment decision.
Finally, for all three investment selection strategies, we compared the paid profit shares, which were based on the predicted initial price, with a scenario where the true initial price is known, resulting in differing payments of the profit shares.

Strategy	Metric	Basis Predicted Initial Price	Basis True Initial Price	Difference
Our Selection	Annual investment growth	0.2786	0.2498	0.0288
	Total investment growth after 5 years	2.4167	2.0499	0.3668
	Absolute investment growth after 5 years	$120,837	$102,494	$18,343
Random Selection	Annual investment growth	0.0794	0.0822	-0.0028
	Total investment growth after 5 years	0.4650	0.4841	-0.0191
	Absolute investment growth after 5 years	$23,252	$24,204	-$952
Perfect Selection	Annual investment growth	0.3240	0.2703	0.0537
	Total investment growth after 5 years	3.0679	2.3081	0.7598
	Absolute investment growth after 5 years	$153,397	$115,403	$37,994

Table 17: Investment Growth for Different Selection Strategies

Comparing the different investment selection strategies, we see that using our machine learning models results in much higher profits. Our method leads to an annual investment growth rate of 27.86%, compared to only 7.94% with a random selection. Even when compared to the perfect selection strategy, which uses future values that aren't known in a real-world scenario, our method still performs well. The perfect selection achieves 32.40% annual growth, which is higher, but not by much compared to the performance of our strategy.
If profit share payments were based on appreciation calculated using true initial prices, our strategy's annual growth would be about 3% lower, and the perfect selection strategy's growth would be around 5% lower. This suggests that we underestimate initial house prices, resulting in higher calculated appreciations. Interestingly, for the random selection strategy, the true annual growth is slightly higher than with the predicted initial prices, indicating that this underprediction doesn't apply to all houses.

However, even when considering the true investment growth, our selection strategy still performs quite well, and the difference compared to the perfect selection strategy becomes even smaller (see Table 17).
When further comparing the actual selected houses in our selection with the perfect selection, we observe that approximately 70% of the houses from the perfect selection are included in our selection (34 houses). This overlap explains the similar performance of the two investment selection strategies.

7.4.4 Exploration of Overpayments and Underpayments

Since we have noted a noticeable difference in the profit share payments depending on whether the predicted initial house prices or the true initial house prices are used to calculate house price appreciation. Therefore, we aim to examine in detail how much more borrowers pay in profit share to the investor compared to what they should pay based on the true appreciation value.

Description	**Value**
Average payment difference	$560.40
Average payment difference in percent	9,13%
Average payment	$3333.61
Avg from the cases borrower payed less (13 cases)	$547.02
Avg from the cases borrower payed more (21 cases)	$1245.95

Table 18: Payment Differences and Averages

Considering the houses from our applied selection strategy, we observe that borrowers pay approximately 9% more to investors than they would need to based on the true appreciation values. On average, borrowers pay a total of $560.40 more than necessary, with the average profit share payment being $3,333.
Focusing on the 34 houses in our selection, we identified 21 cases where borrowers overpaid. In these cases, the average overpayment was $1,245. Conversely, there were 13 cases where borrowers were underpaid, with an average underpayment of $547.02 (see Table 18).
As a result, we can conclude that neither in frequency nor in amount do the cases where borrowers underpay compensate for the cases where they overpay, resulting in an imbalance to the disadvantage of the borrowers.

8 Discussion

8.1 Summary

Our study revealed that the best performance for predicting house prices was achieved using feature extraction techniques such as PCA and clustering in combination with the proposed hybrid model architecture, which uses KNN and multiple XGBoost models. Among the machine learning models tested, XGBoost emerged as the best performer, considering its computational efficiency. We also found that directly predicting price appreciation leads to poor predictive performance. Instead, predicting appreciation indirectly through forecasting the future house price with the predicted current house price as an additional feature and then manually calculating the appreciation provided much better performance.
Key insights into feature importance showed that the living area was consistently one of the top variables across all methods used, reinforcing its critical role in price prediction. Additionally, economic factors such as household income and the HPI were essential predictors of both initial house prices and future appreciation, emphasizing the multifaceted nature of property valuation.
Key findings from our research indicate that machine learning-based investment strategies significantly outperform random selection. The annual investment growth rates for machine learning models were 27.86%, compared to 7.94% for random selection and 32.40% for perfect selection. Despite these promising results, we observed a tendency to underpredict the initial house price. This discrepancy led to an overall overprediction of the appreciation, resulting in borrowers paying approximately 9% more than necessary in profit share payments due to the initial price underestimation.

8.2 Limitations

Our study encountered several limitations. Firstly, we only considered data from one state, Connecticut, and an additional county in Virginia rather than the entire USA. Consequently, model performance may decline if applied to states or counties not included in the training data. Additionally, our dataset included a limited number of core features describing houses. Many other potentially useful features, such as the presence of a garage or the type of roof, were not used due to high levels of missing values, making imputation impractical.
Another limitation was the precision of geocoordinate data. We often matched house data with OpenStreetMap geocoordinates at the street level rather than the house number level, especially on long streets. This imprecision affected the

accuracy of calculated distance and frequency features. Furthermore, we grouped multiple amenities into broad categories, which may have impacted predictive performance. Testing different, more granular groupings was beyond the scope of this study.
We relied entirely on OpenStreetMap for geo-coordinates and amenity data, without the ability to verify its completeness or quality due to the lack of alternative accessible data sources. As an open-source product, its data might be less complete and of lower quality than commercial sources. Moreover, our economic data only spanned from 1990 onwards, with earlier transaction data being imputed, which may have affected predictive performance. Additionally, economic data was only available at the county level, whereas more granular city-level data could have provided richer information.
Our dataset included residential and commercial properties, leading to a higher spread in features such as the number of bathrooms or bedrooms. A more specialized dataset focusing on residential properties might yield better model performance. We primarily focused on tree-based models, not considering deep neural networks, which, despite their complexity, might offer different performance characteristics. Our business model evaluation assumed the true future house price would be known, which is not always the case. In instances where the house is not sold post-lending, we would rely on predicted future prices.
We considered one test scenario with various investments to provide a holistic view of the business model's performance. Other scenarios, which could be examined in future research, might yield different insights. Additionally, we did not discuss the potential interest of borrowers or investors in this business model, which is necessary to understand its general viability.

8.3 Future Research

Future research could explore deep learning techniques and their potential benefits. Additionally, incorporating multi-modal approaches, such as images and textual descriptions of houses, could enhance predictive models. Extracting information from these unstructured data sources using convolutional neural networks and language models like BERT would be valuable, although accessing large datasets of such information is challenging. Considering transactions from across the USA would provide a more comprehensive dataset, albeit requiring significant computational power.

8.4 Contribution to Theory

Our research advances financial innovation by proposing a novel business model that integrates financial innovation with machine learning techniques. By bridging finance, machine learning, and housing economics, we offer a foundation for future business models that build upon or extend our ideas.

8.5 Managerial Implications

Our research provides actionable insights for managers and investors. We demonstrated that machine learning models can closely predict true house prices, fostering trust among investors and borrowers. We identified a slight overprediction of house value appreciations, which can be mitigated by adjusting business model parameters, making it more attractive for borrowers. For instance, reducing the profit share required from the borrower or having investors cover intermediary fees could address this issue. Future research and practical evaluations involving potential borrowers and investors will help determine the best adjustments to enhance the business model's appeal.
Overall, while our study highlights the potential of integrating machine learning with real estate investments, it also underscores the need for further research and practical adjustments to fully realize this potential.

9 Conclusion

In conclusion, this thesis investigates the feasibility of a novel business model that merges property appreciation sharing with machine learning techniques, offering homeowners an alternative financing method while enabling investors to benefit from property value appreciation. The study's primary aim was to explore whether this model could present a viable alternative to traditional financial systems by offering both an attractive investment opportunity for investors and a less debt-intensive financing option for homeowners. The findings indicate a strong potential for success.

A significant result of this research is that the machine learning models effectively predicted both house prices and future appreciation, providing investors with more accurate decision-making tools. From a business perspective, the study confirmed that using machine learning to guide investment selection strategies resulted in much higher returns than random selection. This demonstrates that data-driven approaches enhance the accuracy of property value predictions and substantially increase investor profitability. By ensuring accurate property value predictions, homeowners only pay appreciation shares based on a price that can be realized in the current market.

However, some challenges were noted. While machine learning models consistently improved outcomes for investors, they also revealed an imbalance where borrowers tended to overpay investors, highlighting potential fairness issues in the model's current form. This suggests the need to adjust business model parameters, such as the fraction of appreciation that homeowners must pay to investors. Limitations, such as the dataset's geographical restriction to Connecticut and Virginia, could be addressed in future research, in combination with the incorporation of unstructured data and the exploration of deep learning techniques.

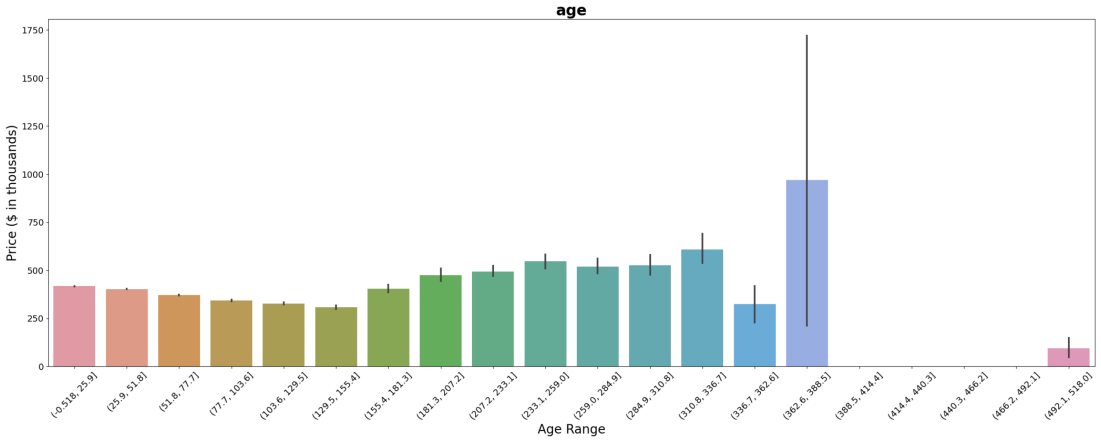
Ultimately, this thesis demonstrates that interdisciplinary approaches combining finance, machine learning, and housing economics offer a compelling solution to real estate challenges. The proposed business model provides a sustainable and equitable alternative to traditional financial systems, offering a promising path forward for both investors and homeowners. By leveraging machine learning, this model could transform how property investments and financing are approached, providing practical tools for improving profitability and accessibility in the real estate market.

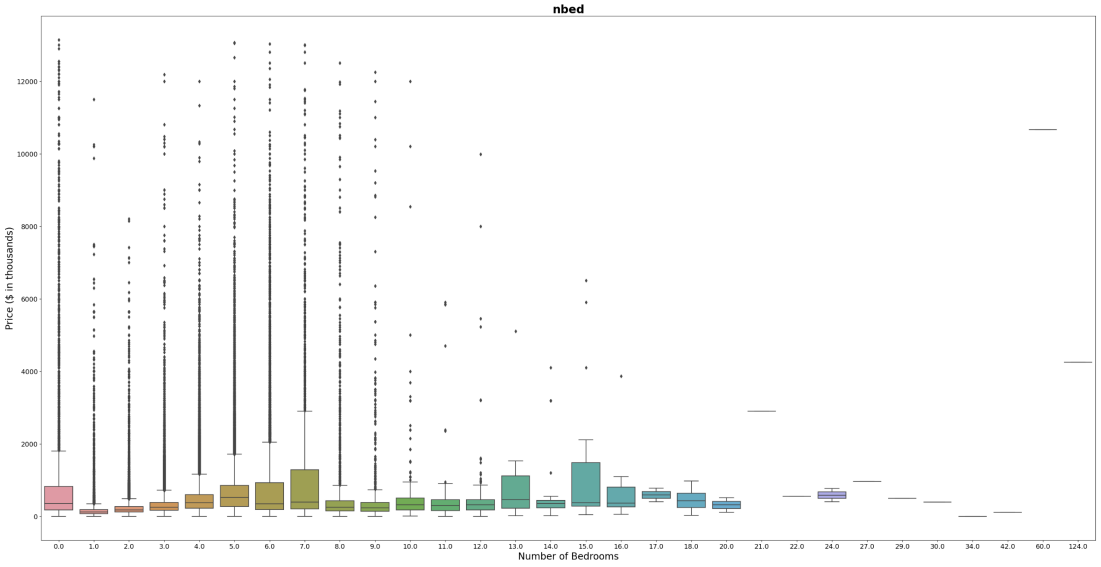
A Appendix

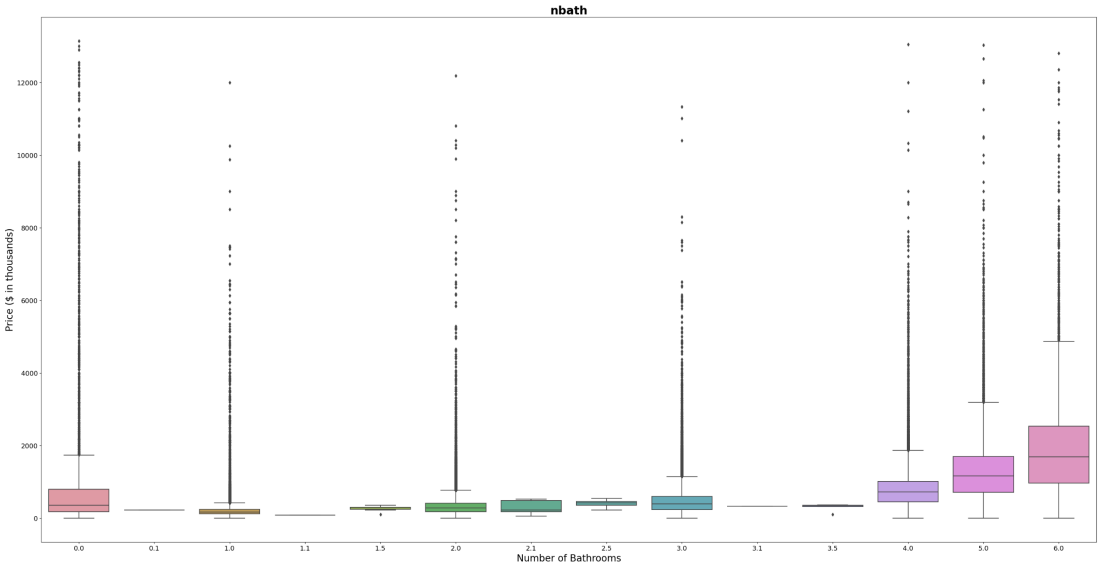
Feature Group	Features
Core Features	city, yrblt, effyrblt, nbed, nbath, nhalfbath, livarea, efflivarea, year, month, age, eff_age, longitude, latitude, county_Fairfax,county_Fairfield, county_Hartford, county_Litchfield, county_Middlesex, county_New Haven, county_New London, county_Tolland, county_Windham, state_Connecticut, state_Virginia, cond_desc_Average, cond_desc_Average_Plus, cond_desc_Fair, cond_desc_Good, cond_desc_Poor
Geo Distance Features	distance_aerodrome, distance_ferry_terminal, distance_railway_station, distance_market, distance_hospital, distance_hotel, distance_museum
Geo Frequency Features	n_reli_inst, n_edu_fac, n_healthcare, n_emergency, n_animalcare, n_commu_venu, n_commu_serv, n_shopping, n_food_drink, n_financial, n_transport, n_entertainment, n_sports, n_utilities, n_adults_entertain,n_accommodation, n_government_civic, n_recreational
Economic Features	hpi, household_income, new_housing, population, n_poverty, poverty_rate, poverty_rate_young, n_poverty_young, n_unemployed, unemployment_rate, civilian_labour, n_employed

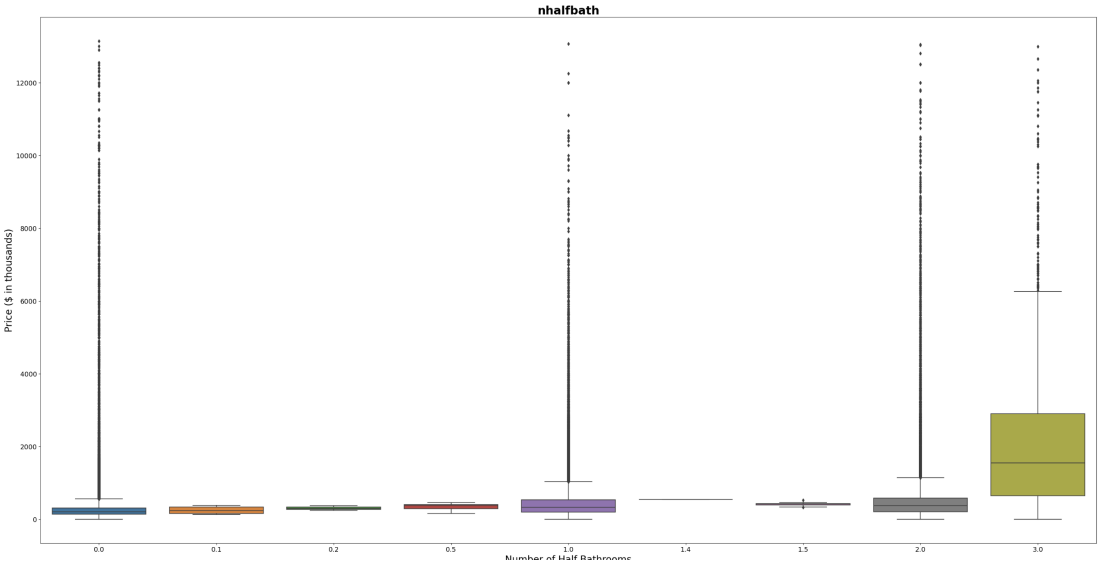
Table 19: Complete List of Features

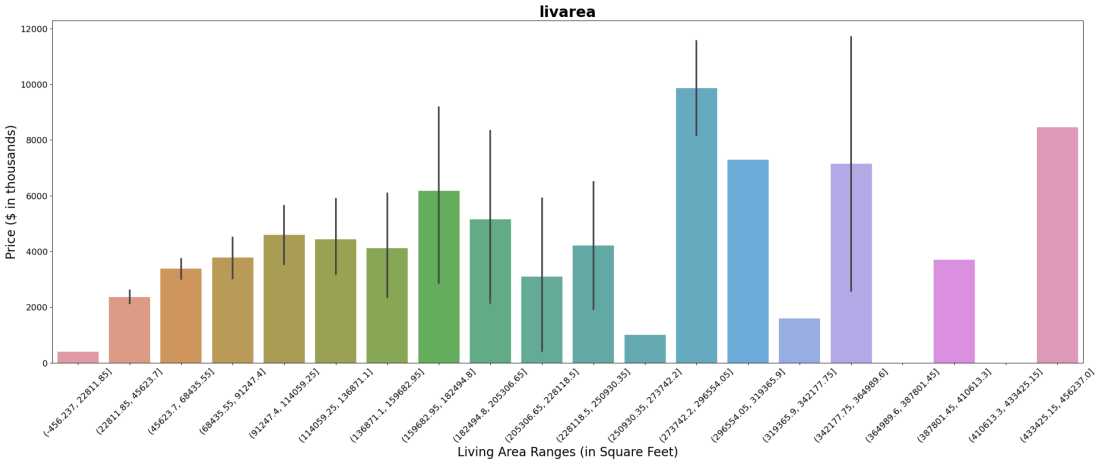
B Appendix

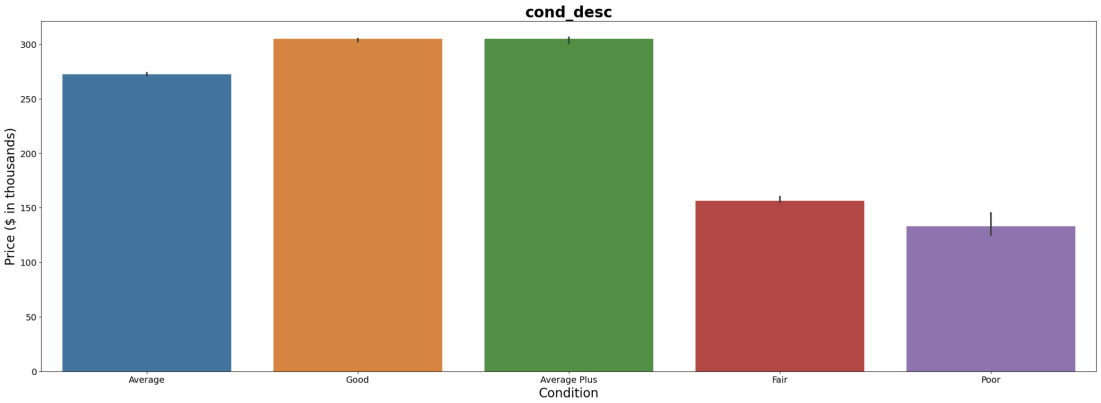

age
Price ($ in thousands)
Age Range
(-0.518, 25.9]
(25.9, 51.8]
(51.8, 77.7]
(77.7, 103.6]
(103.6, 129.5]
(129.5, 155.4]
(155.4, 181.3]
(181.3, 207.2]
(207.2, 233.1]
(233.1, 259.0]
(259.0, 284.9]
(284.9, 310.8]
(310.8, 336.7]
(336.7, 362.6]
(362.6, 388.5]
(388.5, 414.4]
(414.4, 440.3]
(440.3, 466.2]
(466.2, 492.1]
(492.1, 518.0]

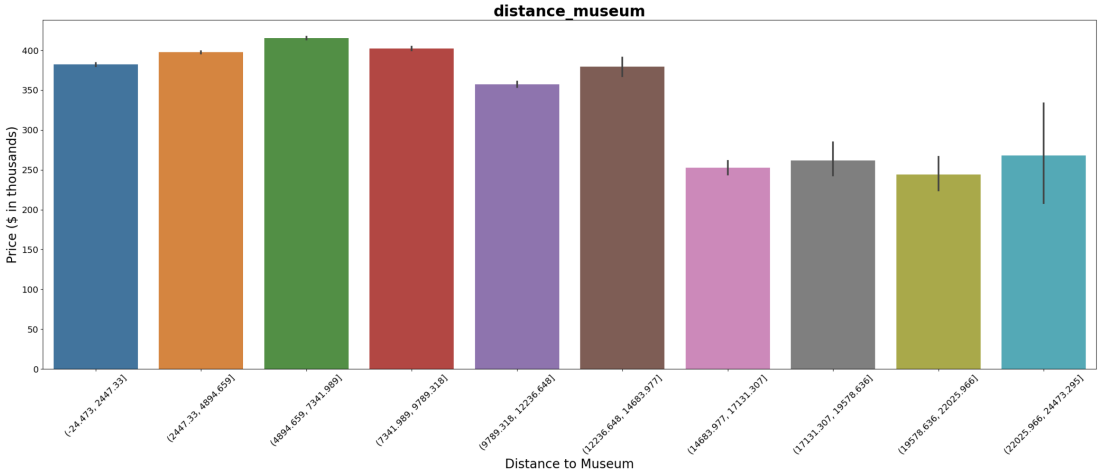

nbed
Price ($ in thousands)
Number of Bedrooms

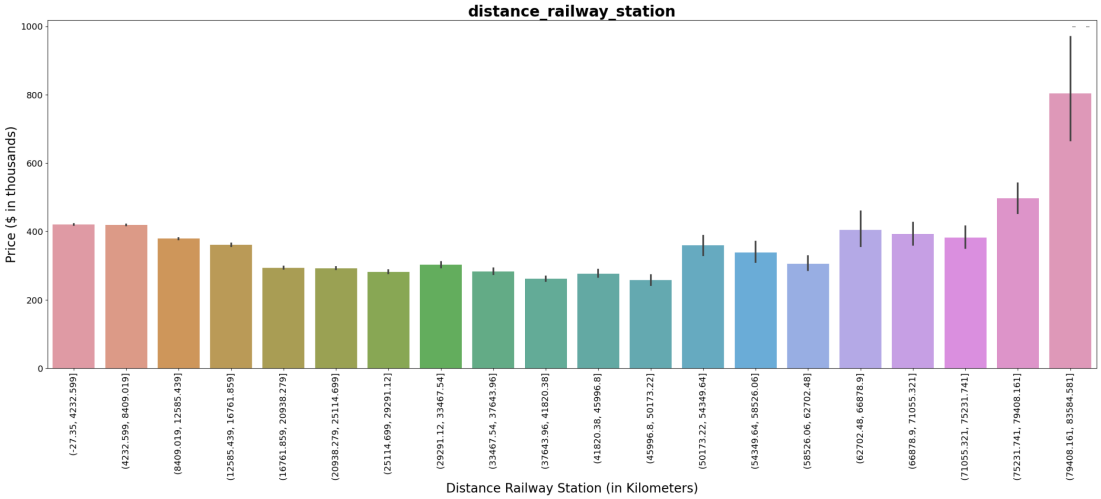

nbath
Price ($ in thousands)
Number of Bathrooms

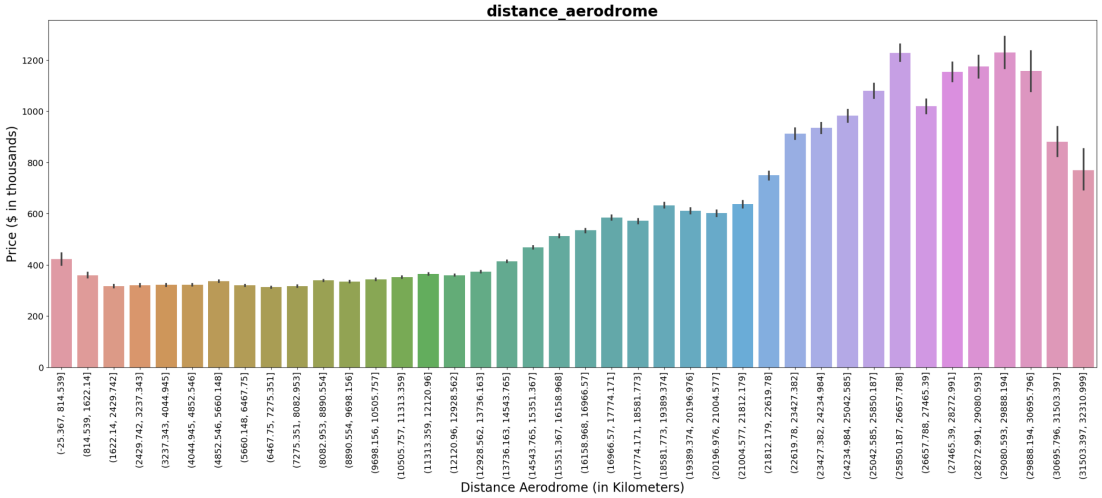

nhalfbath
Price ($ in thousands)


livarea
Price ($ in thousands)
Living Area Ranges (in Square Feet)

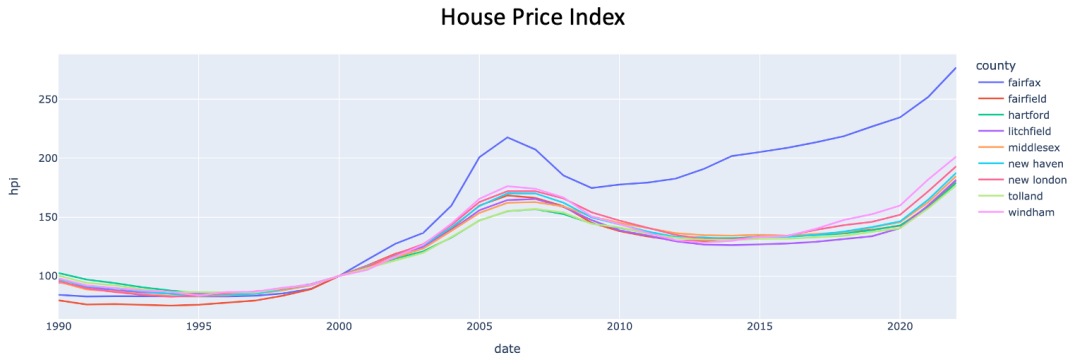

cond_desc
Price ($ in thousands)
0
50
100
150
200
250
300
Average
Good
Average Plus
Fair
Poor
Condition


distance_museum
Price ($ in thousands)
0
50
100
150
200
250
300
350
400
(-24.473, 2447.33]
(2447.33, 4894.659]
(4894.659, 7341.989]
(7341.989, 9789.318]
(9789.318, 12236.648]
(12236.648, 14683.977]
(14683.977, 17131.307]
(17131.307, 19578.636]
(19578.636, 22025.966]
(22025.966, 24473.295]
Distance to Museum


distance_railway_station
Price ($ in thousands)
0
200
400
600
800
1000
(-27.35, 4232.599]
(4232.599, 8409.019]
(8409.019, 12585.439]
(12585.439, 16761.859]
(16761.859, 20938.279]
(20938.279, 25114.699]
(25114.699, 29291.12]
(29291.12, 33467.54]
(33467.54, 37643.96]
(37643.96, 41820.38]
(41820.38, 45996.8]
(45996.8, 50173.22]
(50173.22, 54349.64]
(54349.64, 58526.06]
(58526.06, 62702.48]
(62702.48, 66878.9]
(66878.9, 71055.321]
(71055.321, 75231.741]
(75231.741, 79408.161]
(79408.161, 83584.581]
Distance Railway Station (in Kilometers)


distance_aerodrome
Price ($ in thousands)
0
200
400
600
800
1000
1200
(-25.367, 814.539]
(814.539, 1622.14]
(1622.14, 2429.742]
(2429.742, 3237.343]
(3237.343, 4044.945]
(4044.945, 4852.546]
(4852.546, 5660.148]
(5660.148, 6467.75]
(6467.75, 7275.351]
(7275.351, 8082.953]
(8082.953, 8890.554]
(8890.554, 9698.156]
(9698.156, 10505.757]
(10505.757, 11313.359]
(11313.359, 12120.96]
(12120.96, 12928.562]
(12928.562, 13736.163]
(13736.163, 14543.765]
(14543.765, 15351.367]
(15351.367, 16158.968]
(16158.968, 16966.57]
(16966.57, 17774.171]
(17774.171, 18581.773]
(18581.773, 19389.374]
(19389.374, 20196.976]
(20196.976, 21004.577]
(21004.577, 21812.179]
(21812.179, 22619.78]
(22619.78, 23427.382]
(23427.382, 24234.984]
(24234.984, 25042.585]
(25042.585, 25850.187]
(25850.187, 26657.788]
(26657.788, 27465.39]
(27465.39, 28272.991]
(28272.991, 29080.593]
(29080.593, 29888.194]
(29888.194, 30695.796]
(30695.796, 31503.397]
(31503.397, 32310.999]
Distance Aerodrome (in Kilometers)

C Appendix


House Price Index
hpi
250
200
150
100
1990
1995
2000
2005
2010
2015
2020
date
county
fairfax
fairfield
hartford
litchfield
middlesex
new haven
new london
tolland
windham

References

Abdallah, S. (2015). Using Text Mining To Analyze Real Estate Classifieds. *arXiv*.

Abu Alfeilat, H., Hassanat, A., Lasassmeh, O., Tarawneh, A., Alhasanat, M., Eyal-Salman, H., & Prasath, S. (2019). Effects of Distance Measure Choice on K-Nearest Neighbor Classifier Performance: A Review. *Big Data*, *7*. https://doi.org/10.1089/big.2018.0175

Alsabti, K., Ranka, S., & Singh, V. (2000). An Efficient K-Means Clustering Algorithm. *Proc First Workshop High Performance Data Mining*.

Berente, N., Bin Gu, Recker, J., & Santhanam, R. (2021). Managing Artificial Intelligence. *MIS Quarterly*, *45*(3), 1433–1450. https://doi.org/10.25300/MISQ/2021/16274

Bourassa, S., Cantoni, E., & Hoesli, M. (2010). Predicting House Prices with Spatial Dependence: A Comparison of Alternative Methods. *Journal of Real Estate Research*, *32*, 139–160. https://doi.org/10.1080/10835547.2010.12091276

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Brynjolfsson, E., & Mitchell, T. (2017). What can machine learning do? Workforce implications. *Science*, *358*(6370), 1530–1534. https://doi.org/10.1126/science.aap8062

Can, A. (1992). Specification and estimation of hedonic housing price models. *Regional Science and Urban Economics*, *22*(3), 453–474. https://doi.org/10.1016/0166-0462(92)90039-4

Carleo, G., Cirac, I., Cranmer, K., Daudet, L., Schuld, M., Tishby, N., Vogt, L., & Zdeborova, L. (2019, March). *Machine learning and the physical sciences*.

Carrillo, P. E., de Wit, E. R., & Larson, W. D. (2014, April). Can Tightness in the Housing Market Help Predict Subsequent Home Price Appreciation? Evidence from the U.S. and the Netherlands. https://doi.org/10.2139/ssrn.2139057

Case, B., Clapp, J., Dubin, R., & Rodriguez, M. (2004). Modeling Spatial and Temporal House Price Patterns: A Comparison of Four Models. *The Journal of Real Estate Finance and Economics*, *29*(2), 167–191. https://doi.org/10.1023/B:REAL.0000035309.60607.53

Castelvecchi, D. (2016). Can we open the black box of AI? *Nature*, *538*, 20–23. https://doi.org/10.1038/538020a

Chicco, D., Warrens, M., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. *PeerJ Computer Science*, *7*, e623. https://doi.org/10.7717/peerj-cs.623

Debt - KFF Health News. (n.d.). Retrieved May 7, 2024, from https://kffhealthnews.org/diagnosis-debt/

de Myttenaere, A., Golden, B., Le Grand, B., & Rossi, F. (2016). Mean Absolute Percentage Error for regression models. *Neurocomputing*, *192*, 38–48. https://doi.org/10.1016/j.neucom.2015.12.114

Dobrev, D. (2012). A Definition of Artificial Intelligence. *Mathematica Balkanica*, *19*, 67–74. Retrieved November 17, 2021, from http://arxiv.org/abs/1210.1568

Fisher, A., Rudin, C., & Dominici, F. (2019). All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. https://doi.org/10.48550/arXiv.1801.01489

Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, *29*(5), 1189–1232. Retrieved May 28, 2024, from https://www.jstor.org/stable/2699986

Fürnkranz, J., Chan, P., Craw, S., Sammut, C., Uther, W., Ratnaparkhi, A., Jin, X., Han, J., Yang, Y., Morik, K., Dorigo, M., Birattari, M., Stützle, T., Brazdil, P., Vilalta, R., Giraud-Carrier, C., Soares, C., Rissanen, J., Baxter, R., & De Raedt, L. (2010). Mean Absolute Error. https://doi.org/10.1007/978-0-387-30164-8_525

Geerts, M., vanden Broucke, S., & De Weerdt, J. (2023). A Survey of Methods and Input Data Types for House Price Prediction. *ISPRS International Journal of Geo-Information*, *12*(5), 200. https://doi.org/10.3390/ijgi12050200

Gu, J., Zhu, M., & Jiang, L. (2011). Housing price forecasting based on genetic algorithm and support vector machine. *Expert Systems with Applications*, *38*(4), 3383–3386. https://doi.org/10.1016/j.eswa.2010.08.123

Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN Model-Based Approach in Classification. In R. Meersman, Z. Tari, & D. C. Schmidt (Eds.), *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE* (pp. 986–996). Springer. https://doi.org/10.1007/978-3-540-39964-3_62

Gupta, V., Mishra, V. K., Singhal, P., & Kumar, A. (2022). An Overview of Supervised Machine Learning Algorithm. *2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART)*, 87–92. https://doi.org/10.1109/SMART55829.2022.10047618

Gyourko, J., & Voith, R. (1992). Local market and national components in house price appreciation. *Journal of Urban Economics*, *32*(1), 52–69. https://doi.org/10.1016/0094-1190(92)90014-C

Homelessness Data & Trends. (n.d.). Retrieved May 7, 2024, from https://www.usich.gov/guidance-reports-data/data-trends

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science (New York, N.Y.)*, *349*(6245), 255–260. https://doi.org/10.1126/science.aaa8415

Kang, Y., Zhang, F., Peng, W., Gao, S., Rao, J., Duarte, F., & Ratti, C. (2021). Understanding house price appreciation using multi-source big geo-data and machine learning. *Land Use Policy*, *111*, 104919. https://doi.org/10.1016/j.landusepol.2020.104919

Kern, C., Klausch, T., & Kreuter, F. (2019). Tree-based Machine Learning Methods for Survey Research. *Survey research methods*, *13*(1), 73–93. Retrieved May 27, 2024, from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7425836/

Kherif, F., & Latypova, A. (2020). Chapter 12 - Principal component analysis. In A. Mechelli & S. Vieira (Eds.), *Machine Learning* (pp. 209–225). Academic Press. https://doi.org/10.1016/B978-0-12-815739-8.00012-2

Kodinariya, T. M., & Makwana, P. R. (2013). Review on determining number of Cluster in K-Means Clustering. Retrieved June 28, 2024, from https://www.semanticscholar.org/paper/Review-on-determining-number-of-Cluster-in-K-Means-Kodinariya-Makwana/1a34936bffe558a380168b790dc37956813514ba

Koktashev, V., Makeev, V., Shchepin, E., Peresunko, P., & Tynchenko, V. V. (2019). Pricing modeling in the housing market with urban infrastructure effect. *Journal of Physics: Conference Series*, *1353*(1), 012139. https://doi.org/10.1088/1742-6596/1353/1/012139

Król, A. (2015). Application of Hedonic Methods in Modelling Real Estate Prices in Poland. https://doi.org/10.1007/978-3-662-44983-7_44

Li, J., Du, X., & Martins, J. R. R. A. (2022). Machine learning in aerodynamic shape optimization. *Progress in Aerospace Sciences*, *134*, 100849. https://doi.org/10.1016/j.paerosci.2022.100849

Li, Y., Branco, P., & Zhang, H. (2023). Imbalanced Multimodal Attention-Based System for Multiclass House Price Prediction. *Mathematics*, *11*(1), 113. https://doi.org/10.3390/math11010113

Lundberg, S., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. https://doi.org/10.48550/arXiv.1705.07874

Macpherson, D. A., & Sirmans, G. S. (2001). Neighborhood Diversity and House-Price Appreciation. *The Journal of Real Estate Finance and Economics*, *22*(1), 81–97. https://doi.org/10.1023/A:1007831410843

Mehrotra, K., Mohan, C., & Preface, S. (1997). Elements of Artificial Neural Nets.

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2012, August). *Foundations of Machine Learning*. MIT Press.

Molnar, C. (2020). *Interpretable machine learning*. Lulu. com. Retrieved May 30, 2024, from https://scholar.google.com/scholar?cluster=2937124762106622870&hl=en&oi=scholarr

Monett, D., & Lewis, C. (2018, August). Getting Clarity by Defining Artificial Intelligence—A Survey. In *Studies in Applied Philosophy, Epistemology and Rational Ethics* (pp. 212–214). https://doi.org/10.1007/978-3-319-96448-5_21

Özöğür-Akyüz, S., Erdogan, B., Yıldız, Ö., & Karadayı Ataş, P. (2022). A Novel Hybrid House Price Prediction Model. *Computational Economics*, *62*, 1–18. https://doi.org/10.1007/s10614-022-10298-8

Park, B., & Bae, J. K. (2015). Using machine learning algorithms for housing price prediction: The case of Fairfax County, Virginia housing data. *Expert Systems with Applications*, *42*(6), 2928–2934. https://doi.org/10.1016/j.eswa.2014.11.040

Pereira, J., Stroes, E. S. G., Zwinderman, A. H., & Levin, E. (2021). Covered Information Disentanglement: Model Transparency via Unbiased Permutation Importance. https://doi.org/10.48550/arXiv.2111.09744

Reasons for Being Uninsured Among Adults. (2020). Retrieved May 7, 2024, from https://www.cdc.gov/nchs/products/databriefs/db382.htm

Rosen, S. (1974). Hedonic Prices and Implicit Markets: Product Differentiation in Pure Competition. *Journal of Political Economy*, *82*(1), 34–55. Retrieved June 13, 2024, from https://www.jstor.org/stable/1830899

Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4th ed.). Retrieved December 1, 2021, from https://www.lehmanns.de/shop/mathematik-informatik/56306764-9781292401133-artificial-intelligence-a-modern-approach-global-edition

Sammut, C., & Webb, G. I. (2011). *Encyclopedia of machine learning*. Springer Science & Business Media. Retrieved June 1, 2024, from https://books.google.com/books?hl=en&lr=&id=i8hQhp1a62UC&oi=fnd&pg=PT29&dq=info:IufbymTDBukJ:scholar.google.com&ots=92jezsiHcO&sig=DpArP4LKtcSgsJgpbu8f2uw25_E

Selim, H. (2009). Determinants of house prices in Turkey: Hedonic regression versus artificial neural network. *Expert Systems with Applications*, *36*(2, Part 2), 2843–2852. https://doi.org/10.1016/j.eswa.2008.01.044

Shlens, J. (2014). A Tutorial on Principal Component Analysis. https://doi.org/10.48550/arXiv.1404.1100

Streaks, J. (2024). Average American Debt in 2024: Household Debt Statistics. Retrieved May 7, 2024, from https://www.businessinsider.com/personal-finance/average-american-debt

Turing, A. M. (1950). Computing Machinery and Intelligence. *Mind*, *59*(236), 433–460. https://doi.org/10.1093/mind/LIX.236.433

Wang, L., Wang, G., Yu, H., & Wang, F. (2022). Prediction and analysis of residential house price using a flexible spatiotemporal model. *Journal of Applied Economics*, *25*(1), 503–522. https://doi.org/10.1080/15140326.2022.2045466

Wang, X., Wen, J., Zhang, Y., & Wang, Y. (2014). Real estate price forecasting based on SVM optimized by PSO. *Optik*, *125*(3), 1439–1443. https://doi.org/10.1016/j.ijleo.2013.09.017

Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a Standard Process Model for Data Mining. Retrieved June 15, 2024, from https://www.semanticscholar.org/paper/CRISP-DM%3A-Towards-a-Standard-Process-Model-for-Data-Wirth-Hipp/48b9293cfd4297f855867ca278f7069abc6a9c24

Wright, S. (1921). Correlation and Causation. *Journal of Agricultural Research*, 557–585.

Wu, H., & Wang, C. (2018). A new machine learning approach to house price estimation. *New Trends in Mathematical Science*, *4*, 165–171. https://doi.org/10.20852/ntmsci.2018.327

Yang, L., Liu, S., Tsoka, S., & Papageorgiou, L. G. (2017). A regression tree approach using mathematical programming. *Expert Systems with Applications*, *78*, 347–357. https://doi.org/10.1016/j.eswa.2017.02.013

Yayar, R., & Demir, D. (2015). Hedonic Estimation of Housing Market Prices in Turkey.

Zhang, S., Li, X., Zong, M., Zhu, X., & Cheng, D. (2017). Learning k for kNN Classification. *ACM Transactions on Intelligent Systems and Technology*, *8*(3), 1–19. https://doi.org/10.1145/2990508

Zhao, Y., Chetty, G., & Tran, D. (2019). Deep Learning with XGBoost for Real Estate Appraisal. *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1396–1401. https://doi.org/10.1109/SSCI44817.2019.9002790