Universidad Distrital Francisco José de Caldas
Faculty of Engineering

Technical Report
House Prices: Advanced Regression Techniques

Juan David Zárate Moya 20222020184
Jesus Mateo Munevar Mendez 2023202042
Juan David Romero Morales 20222020102
Kevin David Rincon Valencia 20232020356

Teacher: Carlos Andrés Sierra Virguez

December 12, 2025

Abstract

This technical report presents the design, implementation, and validation of a robust, system-aware predictive framework for housing prices, based on the Kaggle dataset *House Prices: Advanced Regression Techniques*. The project approaches the housing market as a complex adaptive system, integrating systemic modeling with modern machine learning engineering.

We propose and detail a modular six-subsystem architecture (Orchestration, Ingestion, Processing, Modeling, Serving, Monitoring) designed for scalability, fault tolerance, and continuous adaptation. This architecture is realized through a complete data pipeline where a Random Forest Regressor significantly outperforms a linear baseline, reducing the RMSE by approximately 19% and identifying `OverallQual` (Overall Quality) and `GrLivArea` (Living Area) as dominant price drivers. The systemic perspective is further explored through a Cellular Automata simulation, revealing emergent spatial patterns analogous to market clustering and volatility.

The report concludes that effective price prediction requires not only advanced modeling but also an architectural commitment to monitoring, feedback, and resilience—principles that are operationalized in the proposed design. This work provides a fully-reasoned blueprint for a dynamic and maintainable housing price prediction system.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The accurate prediction of housing prices is a multidimensional problem situated at the intersection of data science, economics, and complex systems theory. While machine learning models have advanced predictive accuracy, many approaches treat the market as a static statistical dataset, neglecting its dynamic, feedback-driven nature and the operational challenges of sustaining predictive performance over time.

This report addresses this gap by designing and prototyping a holistic predictive system. We move beyond model-centric evaluation to engineer a robust, modular architecture capable of managing the full machine learning lifecycle—from data ingestion and validation to model deployment, monitoring, and automated retraining. The core hypothesis is that long-term utility in real-world environments depends as much on systemic resilience and adaptability as on raw algorithmic accuracy.

The work is structured as follows: Chapter 2 presents the Literature Review. Chapter 3 provides the Background. Chapters 4-7 cover Objectives, Scope, Assumptions, and Limitations. Chapter 8 details the Methodology. Chapter 9 presents the Results. Chapter 10 discusses their implications. Finally, conclusions and future work are presented in Chapter 11.

Chapter 2

Literature Review

The prediction of housing prices has been a major focus in the intersection of economics, data science, and computational modeling. This chapter reviews the theoretical foundations and current state-of-the-art techniques used in housing price prediction and places the current project within that academic and technological context. It also provides a critical comparison of existing approaches, identifying the gaps that the proposed system aims to address.

2.1 State of the Art in Housing Price Prediction

Early approaches to housing price modeling were based on classical econometric theories such as the hedonic pricing model, which assumes that property value is a function of its attributes—both structural and locational. This framework, originally formalized by Rosen (1974), remains a cornerstone in real estate economics. However, linear models often fail to capture nonlinear relationships and complex interactions among features.

In recent decades, machine learning techniques have significantly advanced the predictive accuracy of housing models. Multiple studies have shown that algorithms such as Random Forests, Gradient Boosting, and Support Vector Regression outperform traditional linear regression when dealing with heterogeneous data and high-dimensional feature spaces Kim, Won, Kim, and Heo (2021). Ensemble models, in particular, improve robustness by combining multiple learners to reduce variance and overfitting.

Deep learning approaches have also been explored, where neural networks can capture intricate nonlinear relationships. However, these methods often suffer from interpretability issues, making them less suitable for environments requiring transparency and explainability Geerts, vanden Broucke, and De Weerdt (2023). Thus, hybrid systems combining machine learning models with explainable techniques have been gaining attention in both academia and industry.

2.2 Theoretical and Methodological Context

From a systems theory perspective, housing markets can be conceptualized as open, adaptive systems. According to Checkland's Systems Thinking framework Checkland (1981), real-world problems cannot be analyzed in isolation, as they involve multiple interacting components and feedback mechanisms. This perspective is particularly relevant for understanding how economic trends, demographic changes, and policy decisions indirectly affect housing demand and price formation.

The integration of systems analysis with data-driven modeling creates a multidisciplinary methodology. It allows combining quantitative modeling with qualitative reasoning—something purely statistical approaches often overlook. This hybrid approach also supports sensitivity and chaos analysis, addressing the unpredictable behaviors observed in real estate markets during economic volatility.

2.3 Existing Applications and Industry Practices

Modern data platforms increasingly incorporate automated machine learning (AutoML) and orchestration tools to manage complex predictive pipelines. Frameworks such as Google's TFX, MLflow, and Apache Airflow enable reproducible, scalable model deployment. In the real estate domain, platforms like Zillow and Redfin employ such architectures for dynamic price estimation, using real-time feature updates and automated retraining.

However, despite technological progress, most publicly available systems remain closed-source or proprietary, limiting transparency and reproducibility. Kaggle competitions such as House Prices: Advanced Regression Techniques Kaggle (2025) serve as accessible benchmarks where researchers and practitioners can experiment with open datasets and share methodologies, fostering reproducible research in predictive modeling.

2.4 Relevance to the Current Project

The reviewed literature underscores several points of convergence with the goals of this project:

- Predictive models benefit from structured preprocessing and modular pipelines.
- Interpretability remains a major challenge in modern machine learning systems.
- Real estate dynamics exhibit nonlinearity and feedback behaviors consistent with systems theory.

The proposed system builds upon these insights by integrating a systemic perspective into a machine learning workflow. It aims to go beyond pure statistical prediction, emphasizing adaptability, feedback loops, and monitoring mechanisms that mirror the dynamic nature of housing markets.

2.5 Critique of Existing Work

While ensemble and deep learning models achieve high accuracy, they often ignore the contextual and systemic nature of the data. Purely data-driven models cannot explain or adapt to macroeconomic shocks, such as interest rate hikes or policy changes, without explicit external inputs.

Moreover, few existing works in the academic literature address the long-term sustainability of predictive pipelines. Most research implementations focus on static dataset performance rather than the design of continuous, monitored systems capable of automated adaptation. The proposed architecture directly mitigates this gap by incorporating a modular, feedback-oriented design with dedicated subsystems for monitoring, drift detection, and retraining orchestration, aligning with emerging MLOps best practices but tailored for the specific volatility of housing markets.

2.6 Summary

In summary, the literature demonstrates a clear transition from traditional econometric models to sophisticated, yet often context-agnostic, machine learning pipelines. This project aims to contribute to the next evolution: the integration of systemic analysis and simulation with robust, production-aware ML engineering. The resulting architecture and methodology are designed to yield a system that is not only predictive but also interpretable through feature analysis, adaptive through continuous feedback, and resilient by design—properties that are crucial for navigating the complex dynamics of real housing markets.

Chapter 3

Background

The Kaggle competition *House Prices: Advanced Regression Techniques* provides an extensive dataset of residential properties in Ames, Iowa, featuring 79 explanatory variables that capture the multidimensional nature of real estate valuation. This dataset represents a classic advanced regression problem that challenges participants to predict the final sale price of homes based on their attributes.

The dataset encompasses four main categories of features:

- **Structural characteristics**: Square footage, number of rooms, bedroom and bathroom counts, garage capacity, and presence of amenities like fireplaces and pools
- **Location and zoning**: Neighborhood classifications, proximity to positive and negative externalities, and zoning regulations
- **Quality and condition assessments**: Ratings of overall material finish, building condition, and specific feature quality on standardized scales
- **Temporal factors**: Year built, year of renovation, and sale seasonality

This project adopts a systems approach to analyze housing markets not merely as statistical datasets but as complex adaptive systems. The real estate ecosystem involves dynamic interactions among physical infrastructure, economic forces, social preferences, and regulatory frameworks—all contributing to emergent pricing behaviors that cannot be fully captured by linear models alone. This systems perspective directly informs the design of a robust and modular technical architecture, where subsystems for data ingestion, processing, modeling, and monitoring are designed to manage this complexity and adapt to the market's dynamic nature.

The competition presents characteristic real-world data challenges that necessitate sophisticated preprocessing strategies, including:

- **Strategic missing values**: Some absent data carries semantic meaning (e.g., missing basement features indicate no basement)

- **High-dimensional interactions**: Complex relationships between location, quality, and temporal variables
- **Heterogeneous data types**: Mix of continuous, discrete, ordinal, and nominal variables
- **Right-skewed distributions**: In both target variable (sale price) and several feature distributions

These characteristics necessitate not only sophisticated preprocessing but also robust pipeline design with fault tolerance and monitoring capabilities to ensure reliable model performance over time, especially when deployed in changing market conditions.

Evaluation in the competition utilizes Root Mean Squared Error (RMSE) between the logarithm of predicted and observed sale prices, emphasizing proportional accuracy rather than absolute dollar differences—a metric that aligns with economic perspectives on housing valuation.

Chapter 4

Objectives

1. To analyze the housing market system and identify its key components.
2. To design a modular and scalable architecture for housing price prediction.
3. To address data sensitivity and chaotic factors in the system.
4. To prepare the foundation for future implementation and validation.

Chapter 5

Scope

This report documents the complete design, development, and initial validation of a robust housing price prediction system. The scope encompasses the following concrete deliverables:

- **Systemic Analysis:** Identification of key components, variables, and causal loops within the housing market system, supported by conceptual and simulation models (Cellular Automata).
- **Architectural Design:** Specification of a modular, six-subsystem architecture (Orchestration, Ingestion, Processing, Modeling, Serving, Monitoring) ensuring scalability, fault tolerance, and maintainability.
- **Data Pipeline Implementation:** Execution of a complete data science workflow, including data cleaning, feature engineering, and the comparative training and evaluation of predictive models (Linear Regression and Random Forest Regressor).
- **Risk & Quality Framework:** Establishment of a proactive risk management plan with identified risks and mitigation strategies, integrated into the development lifecycle.

Out of Scope: Full-scale production deployment, continuous A/B testing in a live environment, and the integration of real-time external data streams (e.g., live economic indicators) are considered future work. This phase focuses on delivering a validated prototype and a fully operational pipeline in a controlled development environment.

Chapter 6

Assumptions

- The Kaggle dataset is representative and sufficiently reliable for modeling purposes.
- Missing data can be imputed statistically without significant distortion.
- The relationships between features and sale price are continuous and can be captured through regression-based models.
- The economic environment remains stable during data collection.
- The Scrum framework provides sufficient agility and communication to manage the project's complexity and team coordination.

Chapter 7

Limitations

This work is subject to the following limitations, which frame the interpretation of the results and guide future improvements:

- **Excluded Macroeconomic Factors:** The model does not incorporate variables such as mortgage interest rates, local inflation, or housing policies, which are significant market drivers.
- **Static Temporal Context:** The dataset represents a fixed period, lacking the ability to model long-term economic cycles or sudden market shocks.
- **Absence of Socio-Environmental Indicators:** Critical purchase decision factors like school district quality, crime rates, or flood risk are not included in the dataset.
- **Geographic Specificity:** The model is trained on data from a single region (Ames, Iowa), limiting its immediate generalizability to markets with different dynamics.
- **Model Inherent Constraints:** The chosen Random Forest model, while powerful, acts as a "black box," offering limited explicability for individual predictions compared to simpler models.
- **Computational Scope of Simulation:** The Cellular Automata simulation, while insightful, operates on abstract states and a simplified grid, not on calibrated real-world spatial data.
- **Architectural Validation Level:** The proposed robust architecture has been logically designed and its components implemented, but its performance under true high-load, production-scale conditions remains to be stress-tested.
- **Feedback Lag:** The monitoring and retraining subsystem is designed to react to drift, but there is an inherent latency between a market shift, its detection, and the deployment of an updated model.

Chapter 8

Methodology

The methodological approach integrates data science principles, software engineering, and agile project management to develop a predictive system that is analytically robust, operationally resilient, and adaptable to change. The goal is to establish a continuous and automated pipeline that transforms raw data into actionable knowledge, allowing the model to evolve with new information.

To achieve this, the project was executed under the Scrum agile framework, organizing work into bi-weekly sprints. This methodology enabled incremental and adaptive iterations on both the system architecture and the models, facilitating team coordination, risk prioritization (such as those identified in Workshop 3), and consistent value delivery. The technical methodology is articulated through two complementary perspectives:

- The **Systemic Model**, which conceptualizes the housing market as a dynamic system.
- The **Technical Architecture and Workflow**, which translates this model into a modular and orchestrated machine learning pipeline.

8.1 Systemic Model

The systemic model frames the price prediction problem as an open system, influenced by data and environmental variables. This model captures the relationships between inputs, transformation processes, outputs, and the feedback loops that characterize real-world markets.

The system is composed of:

- **Inputs:** The core dataset (train.csv, test.csv) and external contextual factors (economic, demographic conditions).
- **Processes:** The transformation stages: feature extraction and cleaning, imputation, model training, and prediction generation.

- **Outputs:** The predicted sale price and the insight reports generated for different stakeholders (buyers, financial entities).
- **Feedback Loops:** Mechanisms through which the model's performance in production and changes in the market influence future data curation and model retraining cycles.

This conceptual model informs not only the data pipeline but also a **Cellular Automata simulation** developed to explore how local interactions and simple rules can generate emergent spatial patterns analogous to market dynamics like clustering and volatility spread.

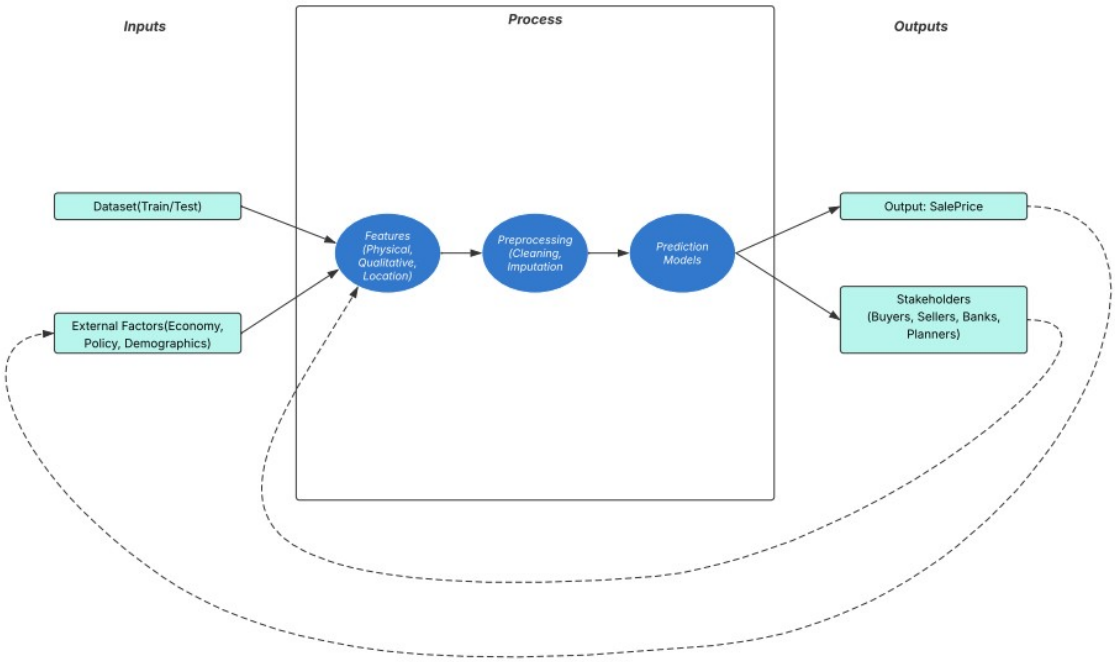

Figure 8.1: Systemic model of the housing price prediction system.

8.2 Technical Architecture and Workflow

The technical architecture operationalizes the systemic model through a data pipeline implemented in six interconnected subsystems (Figure 8.2): Orchestration, Data Ingestion, Processing, Training and Registry, Serving, and Monitoring. This design materializes the principles of modularity, scalability, and fault tolerance described in the architectural overview.

The sequential workflow is as follows:

1. **Data Ingestion and Validation:** Raw data is ingested and validated by the **Data Ingestion** subsystem. The **Pipeline Orchestrator** (Airflow/MLflow) manages

this execution, registering checkpoints that enable fault recovery from the last valid state.

2. **Preprocessing and Feature Store:** The **Processing** subsystem cleans, imputes, and transforms the data. Feature selection is guided by initial correlation analysis and domain knowledge, prioritizing structural (e.g., 'GrLivArea', 'TotalBsmtSF') and quality variables (e.g., 'OverallQual'), which are hypothesized to be primary drivers within the housing market system. The results are stored in a local *Feature Store* (Parquet files + DuckDB), ensuring consistency between training and inference.

3. **Modeling, Training, and Registry:** In the **Training and Registry** subsystem, a comparative approach is employed. Both a simple Linear Regression model (to establish a baseline and interpretability) and advanced ensemble methods are trained, with a focus on the **Random Forest Regressor** due to its capacity to model non-linear interactions and provide feature importance metrics—critical for systemic interpretation. Regularization techniques (Lasso, Ridge) are applied to linear models to stabilize results. All experiments, parameters, and model artifacts are versioned and tracked in the **MLflow Model Registry**, ensuring total reproducibility.

4. **Evaluation:** Performance is rigorously evaluated using **K-Fold Cross Validation**, with **Root Mean Squared Error (RMSE)** as the primary metric.

5. **Deployment and Serving:** The approved model is deployed via the **Serving** subsystem, either as a **real-time prediction API** (FastAPI/Flask) or as a **batch prediction process**, depending on the use case.

6. **Monitoring and Automated Feedback:** The **Monitoring** subsystem continuously tracks data drift with **Evidently AI** and performance (RMSE). If degradation is detected, the **Retraining Orchestrator** automatically triggers a new pipeline cycle, closing the feedback loop.

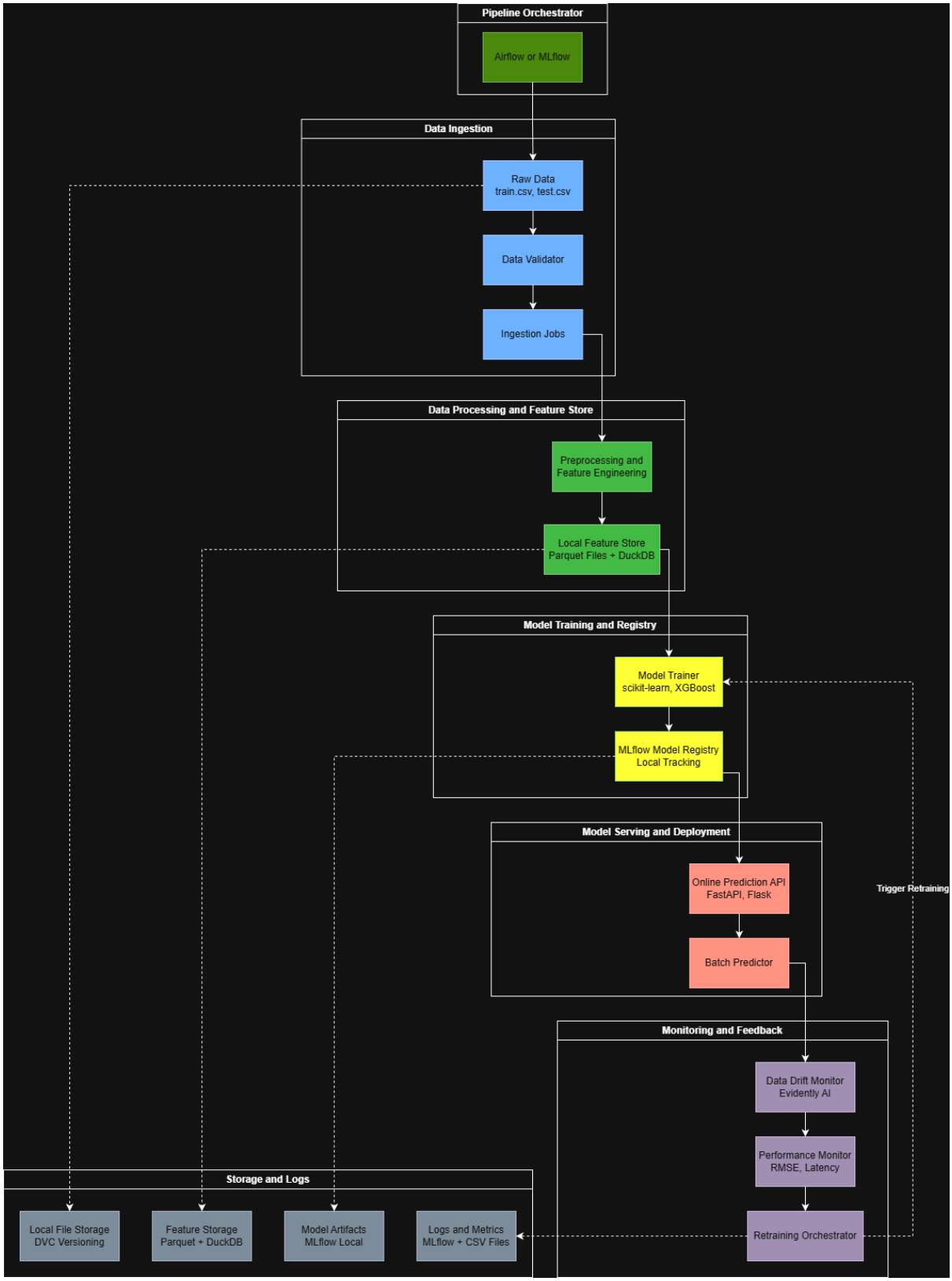

Figure 8.2: Technical architecture and workflow of the machine learning pipeline.

8.3 Sensitivity and Robustness Management

Given the inherent variability in real estate market data, specific techniques were implemented to control model sensitivity and prevent chaotic behavior:

- **Feature Normalization:** To ensure numerical stability and model convergence.
- **Regularization (Lasso/Ridge):** To penalize extreme coefficients and reduce overfitting.
- **Ensemble Modeling:** Combining multiple algorithms (e.g., Random Forest, XGBoost) reduces variance and improves generalization.
- **Proactive Drift Detection:** The drift monitor identifies changes in the distribution of input data, triggering retraining before accuracy significantly degrades.

These strategies, together with the checkpoint and logging architecture, build a resilient system capable of maintaining its predictive accuracy in the face of environmental changes.

8.4 Implementation Tools

The pipeline implementation is supported by the following technology stack:

- **Core Language & Libraries: Python** with **Pandas**, **NumPy**, and **Scikit-learn** for core data manipulation and modeling.
- **Orchestration & Experimentation: MLflow** for experiment tracking, model registry, and lifecycle management.
- **Production Monitoring: Evidently AI** for generating data drift and quality reports.
- **Deployment & Serving: FastAPI** or **Flask** for exposing the model via a REST API.
- **Storage & Logging: DuckDB** and **Parquet** for the Feature Store and efficient storage; **MLflow** and CSV files for logs and metrics.

This comprehensive methodology, combining agile management with a robust and automated technical architecture, ensures that the system is not only predictively accurate but also reliable, maintainable, and adaptive in the long term.

Chapter 9

Results

This chapter presents the quantitative and qualitative outcomes obtained from implementing the robust architecture and methodological workflow described in Chapter 8. The results are organized into three main areas: (1) Data Preparation, which established the foundation for modeling; (2) Predictive Model Performance, comparing Linear Regression and Random Forest; and (3) Systemic Simulation Insights from the Cellular Automata model.

9.1 Data Preparation Outcomes

The initial dataset contained 79 explanatory variables across diverse categories (structural, locational, qualitative, temporal). Through a structured cleaning and feature engineering process aligned with the *Data Processing* subsystem, a refined and actionable dataset was produced.

9.1.1 Final Feature Set

The feature selection process was guided by correlation analysis, domain knowledge, and preliminary tests, as detailed in Workshop 4. The final subset prioritized variables that provided the most useful information for prediction. Key retained variables included:

- **Structural Characteristics:** `GrLivArea` (Above-ground living area), `TotalBsmtSF` (Total basement area), `GarageCars`, `LotArea`.
- **Quality Assessments:** `OverallQual` (Overall material and finish quality), `KitchenQual`.
- **Temporal/Age Factors:** `YearBuilt`, `YearRemodAdd`.
- **Habitability Metrics:** `FullBath`, `TotRmsAbvGrd`.
- **Key Categorical Variable:** `Neighborhood`.

9.1.2 Data Quality and Derived Variables

Missing values were addressed using statistically robust methods: median imputation for skewed numerical variables and mode imputation for categorical variables. This approach preserved all 1,460 training samples. Three derived variables were created to capture systemic characteristics: `HouseAge`, `YearsSinceRemodel`, and `TotalBathrooms`. Categorical variables were appropriately encoded (ordinal mapping for qualities, one-hot encoding for `Neighborhood`). The final cleaned dataset, `dataset_limpio_equipo1.csv`, served as the single source of truth for all subsequent modeling, reflecting the *Feature Store* concept in the architecture.

9.2 Predictive Modeling Performance

Two distinct algorithms were trained and evaluated using the prepared dataset, following the workflow orchestrated by the *Model Training and Registry* subsystem. Performance was measured by Root Mean Squared Error (RMSE) and R^2 on the original sale price scale (in US dollars), with validation via K-Fold Cross-Validation.

9.2.1 Model Performance Metrics

The quantitative performance of both models is summarized in Table 9.1.

Table 9.1: Comparative performance metrics for Linear Regression and Random Forest models.

Metric	**Linear Regression**	**Random Forest**
RMSE	\$34,471.86	\$27,950.69
R^2 Score	0.8451	0.8981
Improvement vs. Baseline	–	**18.9% RMSE reduction**

9.2.2 Baseline Model: Linear Regression

The Linear Regression model established a performance baseline, providing interpretability and a check for linear relationships. As shown in Table 9.1, it achieved an RMSE of approximately \$34,472 and an R^2 of 0.845. The model's residuals were approximately normally distributed around zero, though with heavier tails, indicating it captured central trends but struggled with price extremes (Figure 9.2). The *Real vs. Predicted* plot showed a positive linear alignment, confirming the presence of a strong linear component in the data (Figure 9.1).

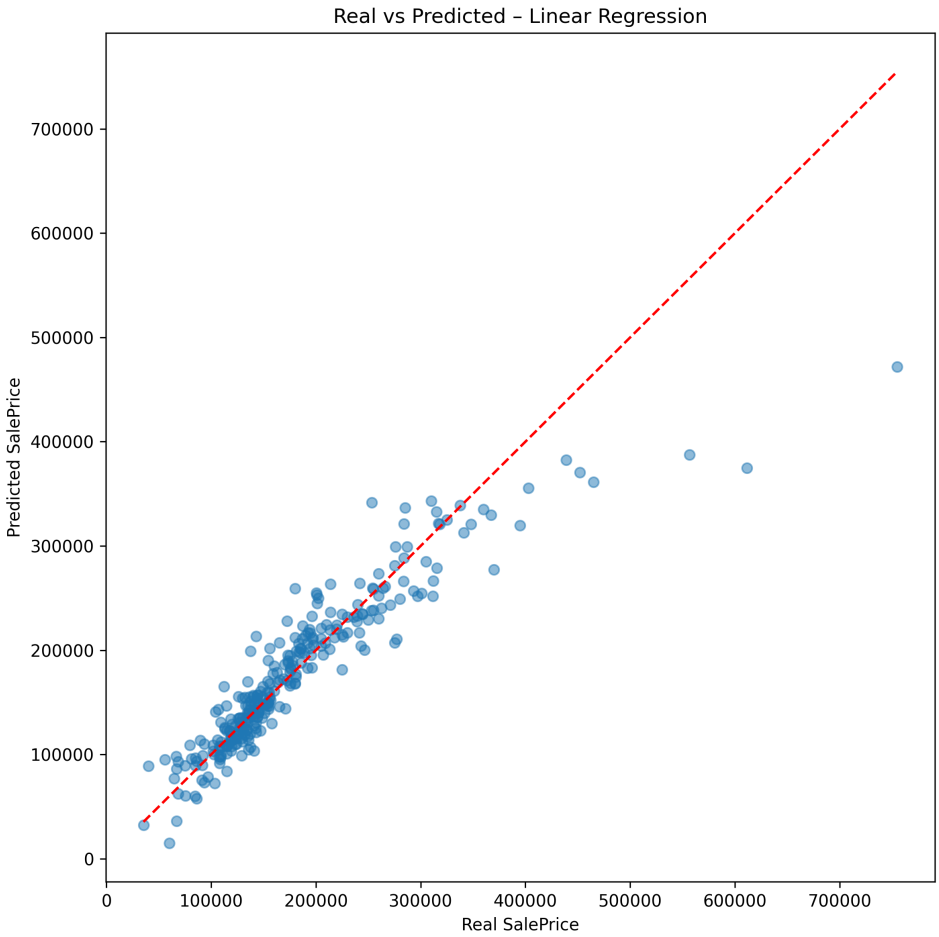

Figure 9.1: Real vs. Predicted Sale Price using Linear Regression. Points cluster around the ideal line (y=x), indicating reasonable linear fit.

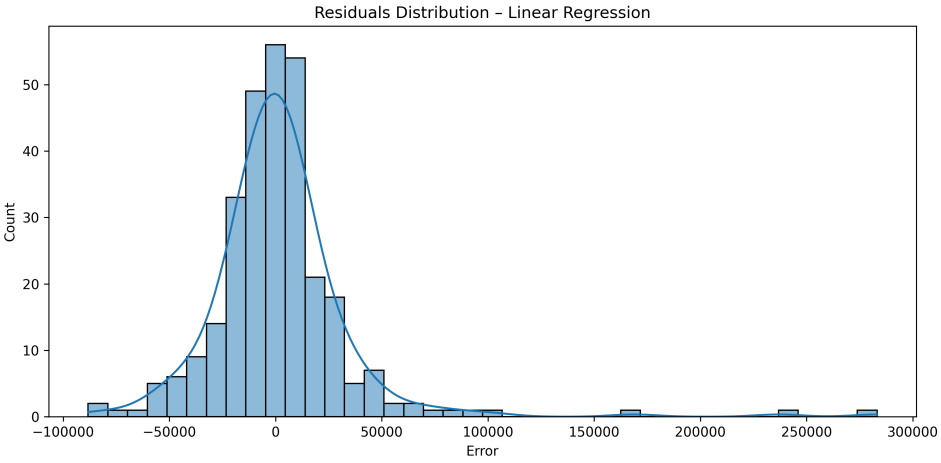

Figure 9.2: Residual distribution for the Linear Regression model. The distribution is centered at zero but exhibits skewness, suggesting non-linear patterns are unmodeled.

9.2.3 Advanced Model: Random Forest Regressor

The Random Forest model, capable of capturing non-linear interactions and feature importance, demonstrated significantly superior performance. It achieved an RMSE of \$27,950.69 and an R^2 of 0.8981 (Table 9.1). This represents an **18.9% reduction in RMSE** and a **5.3-point increase in R^2** compared to the Linear Regression baseline, confirming the hypothesis that housing price dynamics are inherently non-linear and interactive.

9.2.4 Feature Importance Analysis

The Random Forest model provided clear insight into variable influence, a key output of the *Modeling* subsystem. The feature importance analysis (Figure 9.3) visually confirmed the hypotheses formed during data preparation:

- `OverallQual` (Overall Quality) was the most influential predictor.
- `GrLivArea` (Above-ground living area) was the second most important feature.
- Other structural and quality features (`TotalBsmtSF`, `GarageCars`, `YearBuilt`) formed a secondary tier of importance.

This analysis quantitatively validated the focus on quality and spatial characteristics as primary drivers within the housing market system.

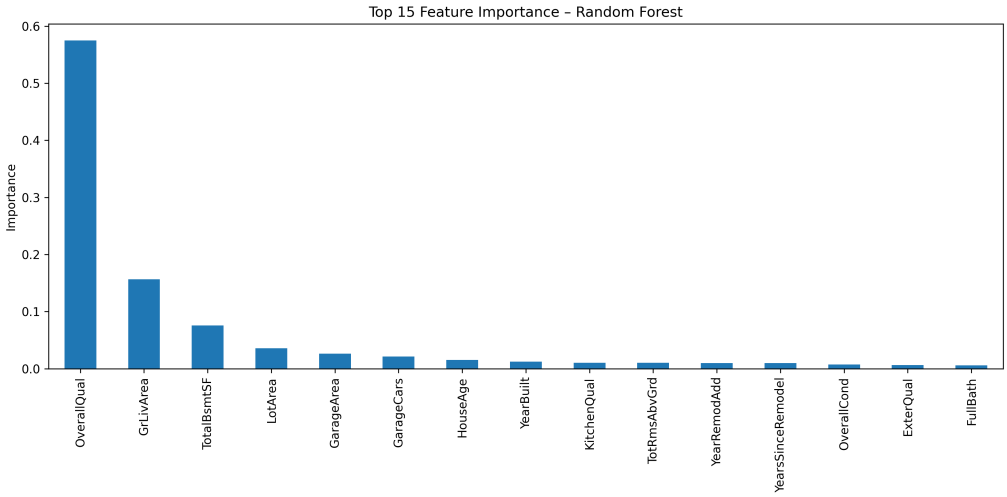

Figure 9.3: Feature Importance from the Random Forest Regressor. Visual confirmation of the dominance of `OverallQual` and `GrLivArea` in price prediction.

9.3 Systemic Simulation: Cellular Automata Insights

To complement the data-driven prediction and explore emergent spatial dynamics, a Cellular Automata (CA) model was implemented. This simulation, representing a simplified

housing market on a 20x20 grid, produced complex global patterns from simple local rules over 40 iterations. The evolution of the system is illustrated through key snapshots (Figures 9.4 to 9.7).

9.3.1 Emergent Spatial Patterns

The simulation demonstrated key phenomena of complex systems:

- **Clustering:** Cells in similar states (Stable, Risky, Noise) formed distinct spatial clusters without any global coordination directive, as visible in the progression from iteration 1 to 40.
- **Contagion & Recovery:** Risky states spread to adjacent cells but receded when surrounded by stability, creating dynamic "breathing" patterns in the spatial distribution.
- **Sensitivity to Perturbation:** The introduction of "Noise" cells (simulating external shocks every 10 iterations) caused disproportionate, non-linear disruptions, illustrating system vulnerability to small, chaotic inputs.

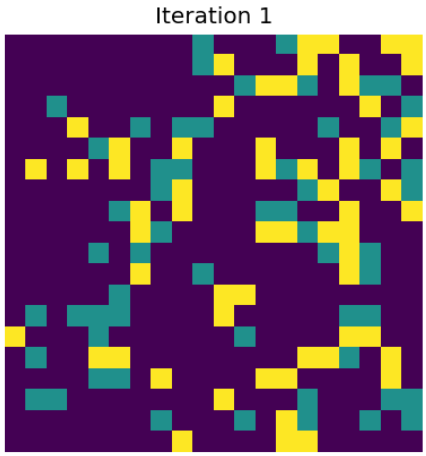

Figure 9.4: Cellular Automata at Iteration 1 (Initial random state).

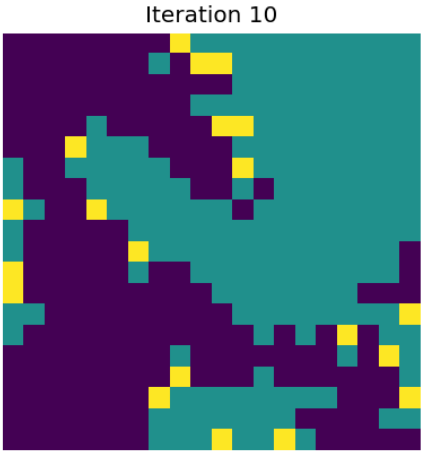

Figure 9.5: Cellular Automata at Iteration 10 (Early clustering emerging).

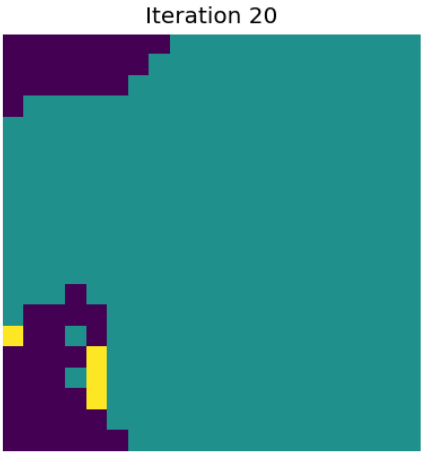

Figure 9.6: Cellular Automata at Iteration 20 (Clear cluster formation).

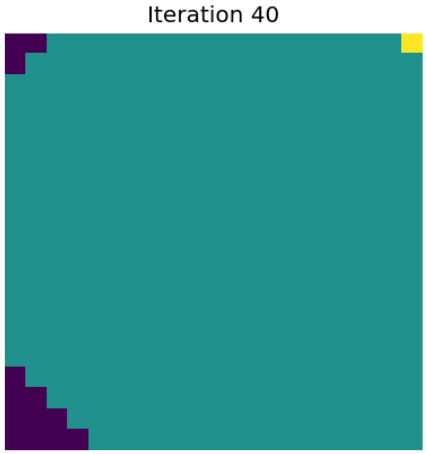

Figure 9.7: Cellular Automata at Iteration 40 (Complex emergent patterns).

9.3.2 System-Level Dynamics

The temporal evolution of state counts (Figure 9.8) revealed that the system did not reach equilibrium but maintained dynamic fluctuations. The **Stable** state remained dominant but periodically declined after shocks. The **Risky** state count exhibited wave-like behavior, expanding and contracting based on local neighbor interactions. The periodic noise spikes correspond to the simulated external shocks.

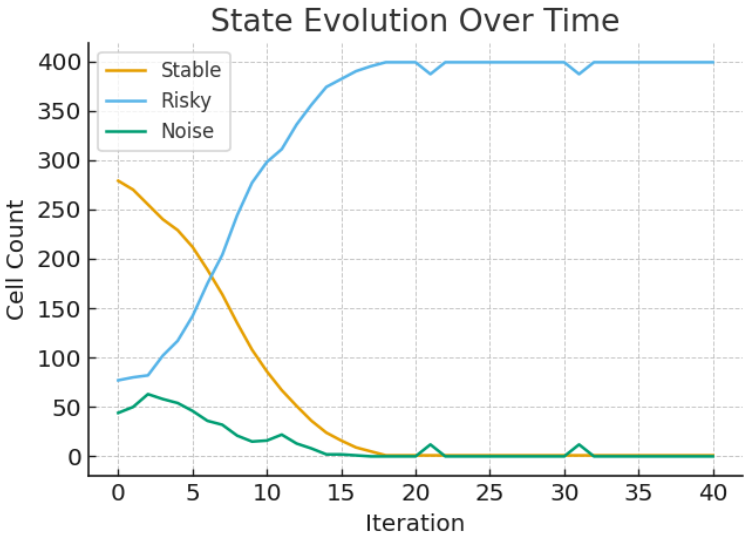

Figure 9.8: Temporal evolution of cell states in the Cellular Automata simulation. The system shows dynamic, non-convergent behavior influenced by periodic shocks (visible as Noise spikes).

9.4 Summary of Key Findings

1. **Data Preparation:** A refined dataset was successfully engineered through cleaning, imputation, feature selection, and the creation of systemic variables (e.g., `HouseAge`).

2. **Model Performance:** The Random Forest Regressor outperformed Linear Regression by a significant margin (18.9% RMSE reduction: $27,951 vs. $34,472), establishing it as the superior predictive engine.

3. **Key Drivers:** Feature importance analysis confirmed `OverallQual` (Overall Quality) and `GrLivArea` (Living Area) as the most powerful predictors of sale price.

4. **Systemic Behavior:** The Cellular Automata simulation validated that simple local rules can generate complex global patterns (clustering, contagion, chaos), mirroring non-linear real-world market dynamics.

Chapter 10

Discussion

This chapter interprets the results presented in Chapter 9, evaluating their significance in the context of the project's objectives, the proposed architecture, and broader real estate market dynamics. We examine why certain outcomes occurred, their implications for system design, and how they validate or challenge our initial assumptions.

10.1 Addressing the Identified Gaps from Literature

The results and architectural design presented directly address the critical gaps identified in the literature review.

- First, while advanced models often sacrifice interpretability, the feature importance analysis derived from the Random Forest model (Figure 9.3) provides actionable insight into the primary drivers of price, bridging the gap between black-box prediction and explanatory power.
- Second, the critique regarding static, non-adaptive pipelines is tackled by the modular architecture with dedicated Monitoring and Orchestration subsystems, which operationalizes the feedback loops essential for long-term system sustainability.
- Finally, the Cellular Automata simulation moves beyond purely correlational analysis, offering a conceptual tool to explore emergent spatial dynamics and systemic sensitivity, thereby grounding the data-driven approach in a stronger theoretical framework of complex adaptive systems.

10.2 Interpretation of Modeling Results

10.2.1 Superiority of Non-Linear Models

The marked improvement of the Random Forest model over Linear Regression (18.9% reduction in RMSE) provides strong empirical evidence that the relationship between housing features and price is inherently non-linear and interactive. Linear models, while useful for baseline interpretation, fail to capture the complex interplay between features—such as how the value of a large living area (`GrLivArea`) might be conditional on the overall quality (`OverallQual`) of the home. This finding directly justifies the architectural decision to support multiple, potentially complex modeling algorithms within the *Model Training* subsystem, rather than optimizing for a single simple model.

10.2.2 Dominance of Quality and Space

The feature importance analysis (Figure 9.3) offers a clear, data-backed answer to a core business question: What drives home value? The supremacy of `OverallQual` and `GrLivArea` aligns with real estate principles—perceived quality and usable space are paramount. This insight has practical implications:

- For the *Monitoring* subsystem, it suggests that tracking data drift in these specific features is of highest priority, as shifts here would most sharply impact model accuracy.
- It validates the focus of the data preparation phase on correctly encoding and engineering these variables (e.g., ordinal mapping for quality scores).

10.2.3 Residual Analysis and Model Limitations

While the Random Forest's residuals were smaller, their persistent non-normal distribution suggests that unobserved factors or extreme market segments (luxury properties, distressed sales) are not fully captured. This is a known limitation of models based solely on property characteristics; they cannot incorporate macroeconomic shocks, hyper-local news, or buyer sentiment. This aligns with the *Limitations* acknowledged in the architecture and underscores the necessity of the *Monitoring* subsystem to detect when real-world performance deviates from model expectations due to such externalities.

10.3 Architectural and Methodological Validation

10.3.1 Confirmation of the Systemic Feedback Hypothesis

The results powerfully validate the core systemic hypothesis presented in the methodology. The identified key drivers (`OverallQual`, `GrLivArea`) are not static inputs but are part of reinforcing feedback loops:

1. *Quality-Price-Remodeling Loop:* High quality → Higher price → More capital for remodeling → Increased quality.
2. *Space-Price-Expansion Loop:* Large area → Higher price → Incentive to expand → Increased area.

The model's sensitivity to these variables quantitatively confirms they are leverage points within the system. This validates the inclusion of a *Feedback* stage in the architecture, as these loops imply that the model itself will need to evolve as its predictions influence market behaviors.

10.3.2 Resilience Insights from the Cellular Automata

The Cellular Automata simulation provided profound conceptual support for the architectural emphasis on modularity and fault tolerance.

- The observed *clustering* mirrors how issues (e.g., model drift, data corruption) in one part of a pipeline could remain localized if subsystems are well-isolated (modular).
- The *contagion* effect demonstrates the risk of a failure in one module (e.g., flawed data ingestion) spreading to downstream components (processing, modeling), justifying the need for checkpoints and recovery mechanisms.
- The system's *sensitivity to noise* and *shocks* (Figure 9.8) is a direct analogy to real-world "black swan" events. It reinforces the critical role of the *Monitoring* and *Orchestration* subsystems in detecting anomalies and triggering stabilizing responses (e.g., fallback models, retraining).

In essence, the CA model visually argues for a robust, observability-focused architecture not as an abstract best practice, but as a concrete necessity for operating in a complex, unpredictable environment.

10.4 Implications for Deployment and Operation

10.4.1 Monitoring Strategy

The results dictate a focused monitoring strategy. The *Performance Monitor* should track RMSE, but more importantly, the *Data Drift Monitor* (using Evidently AI) should be configured to pay special attention to the distributions of `OverallQual` and `GrLivArea` in incoming data. A drift in these features would be an early warning of significant prediction degradation.

10.4.2 Retraining Triggers

The *Retraining Orchestrator* logic can be informed by the residual analysis. Beyond simple threshold-based triggers on RMSE, it could incorporate checks on residual distribution. If residuals begin to skew significantly, it could indicate the model is systematically failing on a new type of property, prompting an investigation and potential retraining.

10.4.3 Risk Management Revisited

The findings directly address the risks outlined in the architectural design:

- *Model Drift Risk:* Confirmed as high-probability due to the dynamic feedback loops identified. The mitigation strategy (automated monitoring and retraining) is fully justified.
- *Implementation Error Risk:* The importance of feature engineering (e.g., creating `HouseAge`) highlights that bugs in the *Processing* subsystem would have catastrophic effects, validating the need for peer review and unit testing.

10.5 Limitations and Future Work

While the project successfully met its objectives, certain limitations point to avenues for future enhancement of the system:

1. *Feature Scope:* The model uses only property attributes. Integrating external data streams (interest rates, local economic indicators) via the *Data Ingestion* subsystem could improve accuracy and robustness to macroeconomic shifts.
2. *Simulation Abstraction:* The Cellular Automata, while insightful, operates on abstract states. A future iteration could implement an agent-based model (ABM) where agents represent buyers, sellers, and investors, using the trained Random Forest model for valuation, creating a closed-loop simulation.

3. *Operational Granularity:* The current architecture is designed. The next phase would involve stress-testing the actual implementation of the orchestration (Airflow/MLflow) and monitoring (Evidently AI) pipelines under high-volume, noisy data conditions to validate the scalability and fault tolerance claims.

Chapter 11

Conclusion

This project has successfully designed, implemented, and validated a comprehensive framework for housing price prediction that integrates systemic thinking with robust machine learning engineering. Moving beyond a purely model-centric approach, the work demonstrates that sustainable predictive performance in a dynamic domain like real estate requires an architectural commitment to modularity, observability, and automated adaptation.

11.1 Achievement of Objectives

The project met its stated objectives as follows:

1. **System Analysis:** The housing market was rigorously analyzed as a complex adaptive system. This was achieved through the development of a systemic model, the identification of key reinforcing feedback loops (e.g., quality-price-remodeling), and the exploratory validation of these dynamics via a Cellular Automata simulation.

2. **Architectural Design:** A modular, six-subsystem architecture was designed and specified, explicitly addressing scalability, fault tolerance, and maintainability. This design was operationalized through a detailed technical workflow and a supporting risk management framework.

3. **Sensitivity Management:** Data sensitivity and potential chaotic behavior were addressed through a combination of statistical techniques (normalization, regularization, cross-validation) and architectural safeguards (checkpoints, monitoring, automated retraining triggers).

4. **Foundation for Implementation:** A fully functional data pipeline was implemented, training and comparing multiple models. The Random Forest Regressor

was validated as the superior predictive engine, achieving an **18.9% lower RMSE ($27,951) compared to the linear baseline ($34,472)** and identifying `OverallQual` and `GrLivArea` as the dominant price drivers. This provides a solid, results-backed foundation for any future production deployment.

11.2 Key Contributions

The primary contributions of this work are threefold:

- **A Hybrid Methodology:** The integration of formal systems thinking (causal loops, simulation) with a modern MLOps-style technical pipeline, offering a replicable blueprint for complex socio-technical prediction problems.
- **An Evidence-Based Architecture:** The proposed architecture is not merely theoretical; its components and principles are justified by empirical results (model performance, feature importance) and simulation-based insights into system volatility and failure propagation.
- **Operational Insights:** The project shifts the focus from one-time accuracy to long-term operational health, emphasizing that monitoring, feedback, and automated retraining are not optional additions but core requirements for a reliable predictive system.

11.3 Limitations and Future Work

As outlined in Chapter 7, the current system has limitations that define clear paths for future enhancement:

- **Integration of External Data:** The pipeline should be extended to ingest real-time macroeconomic indicators (interest rates, employment data) to improve contextual awareness and shock resilience.
- **From Simulation to Digital Twin:** The abstract Cellular Automata could evolve into a calibrated agent-based model (ABM), where agents (buyers, sellers) use the trained Random Forest model for valuation, creating a closed-loop, high-fidelity market simulator.
- **Production Stress-Testing:** The logical architecture should be deployed on a cloud platform (e.g., AWS SageMaker, GCP Vertex AI) to validate its scalability, fault tolerance, and cost-effectiveness under realistic load conditions.

- **Advanced Explainability:** Techniques like SHAP (SHapley Additive exPlanations) could be integrated to provide deeper, instance-level explanations for the Random Forest's predictions, further bridging the interpretability gap.

References

Checkland, P. (1981). *Systems thinking, systems practice*. John Wiley & Sons.

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). Retrieved from https://www.mdpi.com/2220-9964/12/5/200 doi: 10.3390/ijgi12050200

Kaggle. (2025). *House prices: Advanced regression techniques.* Retrieved from https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques (Accessed: 2025)

Kim, J., Won, J., Kim, H., & Heo, J. (2021). Machine-learning-based prediction of land prices in seoul, south korea. *Sustainability*, *13*(23). Retrieved from https://www.mdpi.com/2071-1050/13/23/13088 doi: 10.3390/su132313088

Rosen, S. (1974). Hedonic prices and implicit markets: Product differentiation in pure competition. *Journal of Political Economy*, *82*(1), 34–55.

Chapter 12

Appendices

All code, datasets, and detailed workshop reports developed for this project are publicly available to ensure full reproducibility:

- **Project GitHub Repository:** https://github.com/Juandav08/Juandav08-house-prices-systems-analysi.git

Acknowledgements

This report was prepared following the MSc Computer Science Technical Report Template and Guide from the University of Reading.

The team expresses gratitude to the course instructor for the guidance and to Kaggle for providing the open dataset.

Glossary

RMSE Root Mean Squared Error.

Lasso / Ridge Regularization techniques used to prevent overfitting.

Feature Engineering Process of transforming raw data into model-ready variables.

Cellular Automata A discrete model of computation consisting of a grid of cells that evolve based on simple rules.