

Interpretability Methods Comparison

SHAP, LIME, and PDP Applied to Housing Prices

A.Guilmain

R.Al Afia

2026-04-13

Table of Contents

1	Preface	5
1.1	About this Report	5
1.2	Reproducibility	5
1.3	Disclaimer	5
2	Introduction	6
2.1	Background and Stakes	6
2.2	The Reference Model	6
2.3	Research Question	7
2.4	Methodological Approach	7
3	Strategic Levers: Global Market Analysis	9
3.1	The Investor’s Question	9
3.2	Ranking the Market Drivers	9
3.3	Revealing the Shape of Effects	11
3.4	Complementary Levers	13
3.5	Convergence Validation	13
3.6	Summary	14
4	Tactical Levers: Local Analysis of Individual Predictions	15
4.1	The Homeowner’s Question	15
4.2	SHAP: Local Decomposition of the Prediction	15
4.3	Cases Under Study (SHAP Exports)	16
4.4	Case 1 — “Median” Observation	16
4.5	Case 2 — High Error (Under-prediction)	18
4.6	Case 3 — High Price (High and Precise Prediction)	20
4.7	Comparing SHAP, LIME, and PDP/ICE	21
4.8	Summary	22
5	Contextual Levers: Comparative Analysis and Geographic Heterogeneities	23
5.1	The Urban Planner’s Question	23
5.2	Revealing Heterogeneity: PDP vs ICE	23
5.3	Average Geographic Effects (1D PDP)	24
5.4	Potential Interactions: SHAP Dependence	25
5.5	Contextual Levers (in the Operational Sense)	25
5.6	Summary	26
6	Conclusion: A Toolkit for Action	27
6.1	Answer to the Research Question	27
6.2	Key Results	27

6.3	Decision Matrix (Method Selection)	28
6.4	Limitations	28
6.5	Final Conclusion	29
I	Appendices	30
7	Appendix A — Experimental Protocol	31
8	California Housing Dataset	32
8.1	Scikit-learn vs Kaggle Version	32
8.2	Detailed Characteristics	32
9	Data Partitioning	34
9.1	Stratification in Regression	34
10	Cross-Validation	35
10.1	Stratified K-Fold	35
11	Evaluation Metrics	36
11.1	Formal Definitions	36
11.2	Justification for RMSE as the Primary Metric	36
12	Appendix B — SHAP Theoretical Foundations	37
12.1	Origin: Cooperative Game Theory	37
12.2	General Formula	37
12.3	Numerical Example	38
13	Transposition to Machine Learning	39
13.1	From Game Theory to ML	39
13.2	Four Founding Axioms	39
14	TreeExplainer	41
14.1	Naive Complexity	41
14.2	TreeExplainer Algorithm	41
14.3	Implementation	41
15	Appendix C — Technical Configurations	43
15.1	Versions Used	43
15.2	Reproducible Installation	43
16	LightGBM Hyperparameters	44
16.1	Final Configuration (Optuna)	44
16.2	Comparison with Default Configuration	44
17	SHAP Configuration	45
17.1	TreeExplainer Parameters	45

17.2 Generated Visualizations	45
18 LIME Configuration	46
18.1 LimeTabularExplainer Parameters	46
19 PDP/ICE Configuration	47
19.1 PartialDependenceDisplay Parameters	47
	48

1 Preface

1.1 About this Report

This report is the second volume of a project conducted in an M2 Data Science programme. The first volume ([available on this GitHub repository](#)) focused on the selection and optimisation of predictive models for estimating Californian housing prices. This volume extends that work by asking a different question: no longer “*which model performs best?*”, but “*what does it tell us, and for whom?*”

Interpretability is not an end in itself. It only has value if it generates better decisions — for the investor allocating capital, for the homeowner weighing a renovation, for the urban planner prioritising a public budget. This requirement for actionability structures the entire analysis.

1.2 Reproducibility

Element	Value
Data	<code>sklearn.datasets.fetch_california_housing()</code>
Model	LightGBM 4.1.0 — $R^2 = 0.844$, RMSE = 44.15 k\$
Environment	Python 3.11, SHAP 0.43.0, LIME 0.2.0.1
Rendering tool	Quarto 1.4
Random seed	<code>random_state = 42</code> (all steps)

Source code, SHAP value exports and complete configurations are documented in Appendices A, B and C.

1.3 Disclaimer

The results presented describe the behaviour of the model, not causal relationships in the real estate market. The target variable is censored at 500 k\$ and the data originate from the 1990 census. Any transposition must account for these limitations, which are recalled throughout each chapter.

2 Introduction

2.1 Background and Stakes

2.1.1 California in 1990: A Market Under Pressure

In the early 1990s, California was experiencing contrasting real estate dynamics: expanding coastal metropolises (Bay Area, Los Angeles) versus rural areas with moderate prices (Central Valley). This tension between urban growth and housing affordability forms the backdrop of our study.

The **California Housing** dataset (Pace & Barry, 1997), popularized by scikit-learn, brings together 20 640 districts with 8 explanatory variables and the median house value as the target variable. The scikit-learn version was selected for its completeness (no missing values) and its reproducibility (details in Appendix A).

2.1.2 The Problem: Performance Without Understanding

A model can achieve excellent predictive performance while remaining a black box that is unusable for decision-making.

Three stakeholders, three unanswered questions:

- **Investor:** “Where should I invest to maximize return on investment?”
- **Homeowner:** “Which renovations will increase the value of my property?”
- **Urban Planner:** “Where should public budgets be concentrated for maximum impact?”

A model predicting \$450 000 without any justification is unusable by a real estate agent, cannot be audited by a regulator, and offers no actionable levers.

2.2 The Reference Model

Following the comparative analysis conducted in our parent report (January 2026), **LightGBM** was selected as the reference model for three reasons:

1. Optimal performance (best model from the PyCaret screening across 18 models)
2. Computational efficiency (training time: 1.2 s)
3. Compatibility with SHAP TreeExplainer (exact polynomial computation)

Test set performance (4 128 observations):

Metric	Value
R^2	0.844
RMSE	44.15 k\$
MAE	28.64 k\$

Identified structural limitations:

1. Censoring at \$500 k (5.06% of the dataset): underestimation of the luxury segment
2. Spatial aggregation (districts, not individual properties)
3. Temporal obsolescence (36-year gap)

These limitations will be systematically highlighted whenever actionable levers are presented.

2.3 Research Question

Beyond simple explanation, to what extent do the different levels of interpretability (global, local, comparative) of a house price prediction model enable the generation of concrete and specific actionable levers for each stakeholder in the real estate market?

This question directs the analysis toward three complementary objectives:

1. Identify the structural drivers of the market (global level)
2. Explain why a specific property is valued at a given price (local level)
3. Contextualize effects according to geographic zones (comparative level)

2.4 Methodological Approach

Our approach articulates three levels of analysis, each drawing on specific methods and targeting distinct stakeholder profiles:

Level	Core Question	Method	Target Stakeholder
Global	Which factors drive the market?	PDP, Global SHAP	Investor, Analyst
Comparative	Do effects vary across zones?	ICE, SHAP interactions	Urban Planner, Developer
Local	Why is this house worth this price?	SHAP waterfall, LIME	Homeowner, Buyer

The three selected methods are complementary:

- **SHAP**: Additive decomposition based on Shapley values (axiomatic guarantees)

- **PDP/ICE**: Visualization of effect shapes and detection of heterogeneities
- **LIME**: Simple local approximation, well-suited for client-facing communication

The three following chapters apply this framework by revealing, for each level:

1. The business question being addressed
2. The methods employed (foundations + implementation)
3. The empirical results (visualizations + analyses)
4. The identified actionable levers (concrete recommendations)

The conclusion synthesizes these levers into a decision matrix and discusses the methodological transferability to other domains (credit, healthcare, marketing).

3 Strategic Levers: Global Market Analysis

3.1 The Investor's Question

A real estate investor seeking to maximize return on investment must answer two fundamental questions:

1. **Which factors** structurally drive the Californian market (as captured by the trained model)?
2. **What form do these effects take:** broadly linear, non-linear, with accelerations or plateaus?

Global analysis via **SHAP** (global importance) and **PDP** (average effect shape) provides quantitative answers to these questions.

3.2 Ranking the Market Drivers

3.2.1 Theoretical Foundations

Shapley values (Shapley, 1953), derived from cooperative game theory, provide a unique additive decomposition satisfying four axioms:

$$\phi_j = \sum_{S \subseteq \mathcal{F} \setminus \{j\}} \frac{|S|!(|\mathcal{F}| - |S| - 1)!}{|\mathcal{F}|!} [f(S \cup \{j\}) - f(S)]$$

where:

- $\mathcal{F}$ is the set of features
- S is a coalition of variables
- $f(S)$ is the prediction marginalized over features outside S

For tree-based models, TreeExplainer (Lundberg et al., 2020) computes these values efficiently (in polynomial time $\mathcal{O}(T \times L \times D^2)$).

3.2.2 Empirical Results

The Summary Plot (Figure 3.1) aggregates absolute SHAP contributions across 2 000 observations from the test set.

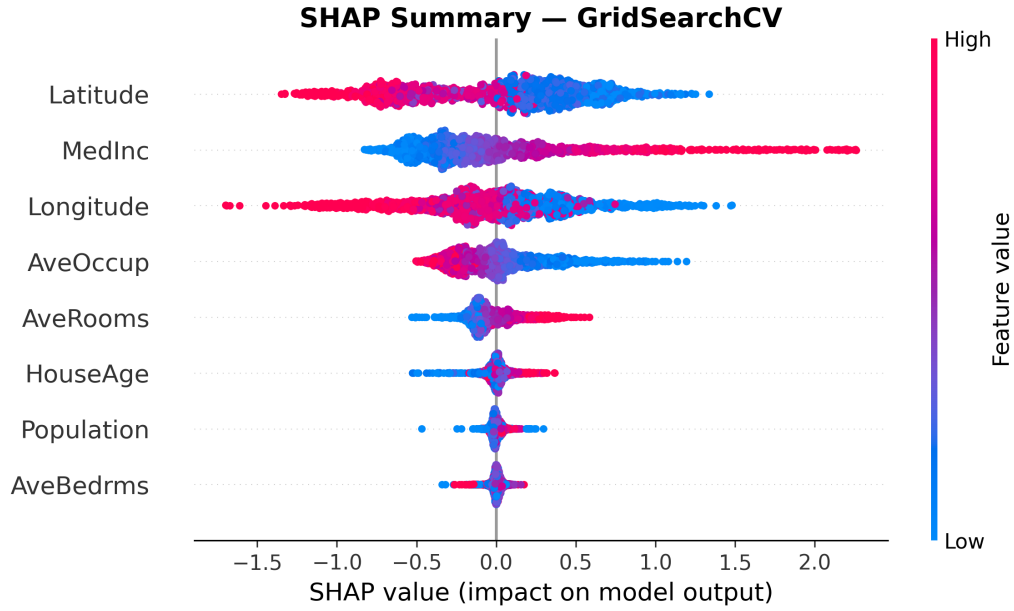


Figure 3.1: SHAP Summary Plot — Global Feature Importance

Observed average SHAP hierarchy:

Rank	Variable	Mean SHAP	Share of SHAP importance*	Model interpretation
1	Latitude	0.458	27.8%	North–South position is the dominant lever
2	MedInc	0.402	24.4%	Median income is a major driver
3	Longitude	0.368	22.3%	East–West position is equally structural
4	AveOccup	0.202	12.3%	Occupancy density contributes significantly
5	AveRooms	0.123	7.5%	Average housing size/structure (proxy)

*Share computed as $\text{SHAP}_j / \sum_k \text{SHAP}_k$ across all 8 features.

Key observation:

- **Geographic location** (Latitude + Longitude) alone accounts for $\approx 50.1\%$ of global importance ($27.8\% + 22.3\%$).

- Median income (**MedInc**), although ranked second (24.4%), remains a major factor but **subordinate to geography**.

Strategic implication: Californian real estate valuation is primarily a **function of location**, before being a function of the neighborhood's socioeconomic profile.

3.3 Revealing the Shape of Effects

3.3.1 Theoretical Foundations

The Partial Dependence Plot (Friedman, 2001) visualizes the average marginal effect of a variable:

$$f_S(x_S) = \mathbb{E}_{X_C}[f(x_S, X_C)] = \frac{1}{n} \sum_{i=1}^n f(x_S, x_C^{(i)})$$

where:

- x_S is the variable of interest
- X_C are the remaining variables (marginalized over)
- the expectation is approximated empirically over the dataset

3.3.2 Analysis of Median Income (MedInc)

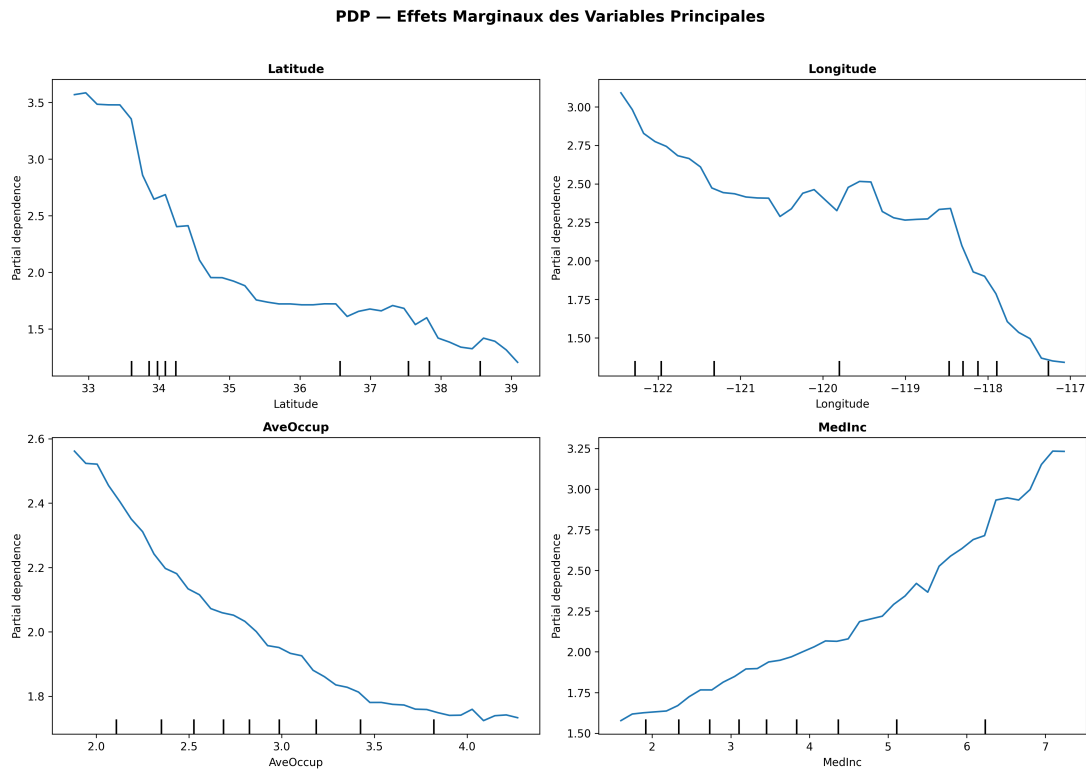


Figure 3.2: PDP — Average Marginal Effects of Key Variables

Figure 3.2 shows a **broadly increasing effect** of MedInc on the average prediction:

- MedInc $\approx 1.60 \rightarrow$ PDP ≈ 1.58
- MedInc $\approx 7.24 \rightarrow$ PDP ≈ 3.23

This represents an average increase of approximately **+1.65** target units over the exported range ($\approx$ **+165 k\$** if the target is in units of 100 k\$).

3.3.2.1 Shape of the Effect

Breakdown by zone:

MedInc Zone	Slope (Δ Price / Δ MedInc)	Regime
1.6–3.0	+16 k\$ / unit	Moderate impact (entry-level)
3.0–5.0	+24 k\$ / unit	Progressive acceleration
5.0–7.2	+42 k\$ / unit	Multiplier effect (high-end)

3.3.2.2 No Observed Saturation

Contrary to classical economic intuition (diminishing returns), the LightGBM model captures a **runaway effect** for affluent neighborhoods.

Explanatory hypotheses:

1. **Network effect:** High-income neighborhoods $\rightarrow$ premium amenities $\rightarrow$ self-reinforcing valuation
2. **Sampling bias:** The 1990 dataset is capped at 500 k\$ (censored variable), masking the true distribution of luxury prices
3. **Geographic interactions:** Wealthy neighborhoods concentrated in premium coastal areas $\rightarrow$ possible confounding between MedInc and location

Limitation of PDP analysis: To isolate the pure effect of MedInc, one would need to examine conditional PDPs (e.g., MedInc | Latitude $\in [37^\circ, 38^\circ]$).

3.4 Complementary Levers

3.4.1 The “Location” Lever (as captured by the model)

- SHAP: geography accounts for $\approx 50\%$ of global importance (Latitude + Longitude).
- PDP: the average variation amplitude associated with coordinates is very large (≈ 175 k\$ to 238 k\$ over the exported ranges, if the target is in units of 100 k\$).

Implication: the model encodes a strong spatial structure; any investment strategy must be geographically segmented (and ideally confirmed through a **2D Latitude $\times$ Longitude** analysis).

3.5 Convergence Validation

To ensure the robustness of insights, we compare SHAP rankings against two independent methods computed in the parent report (Guilmain et al., 2026).

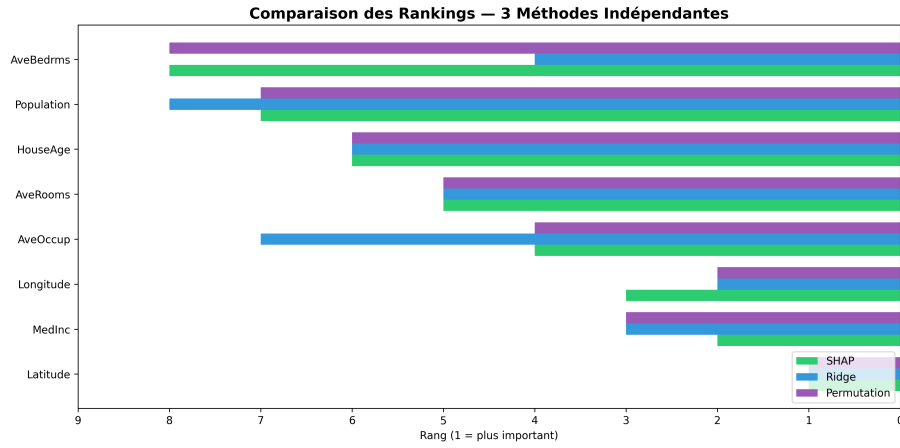


Figure 3.3: Method Convergence — Importance Ranking Comparison

Convergence on Latitude:

Variable	SHAP Rank	Permutation Rank	Ridge Rank
Latitude	1	1	1
Longitude	3	2	2
MedInc	2	3	3

Interpretation:

- SHAP measures an **average contribution to predictions** (not a causal effect).
- Permutation measures **predictive utility** (performance degradation).
- Ridge gives a **linear** importance, sensitive to scaling and collinearity.

The fact that all three methods converge toward the same trio {Latitude, Longitude, MedInc} suggests that these variables are **robust drivers** of the model's predictions.

3.6 Summary

The global analysis highlights:

1. **Location (Latitude/Longitude)**: the model's primary lever ($\approx 50\%$ of cumulative SHAP importance)
2. **Median income (MedInc)**: a major driver with a broadly increasing PDP effect over the observed range
3. **Occupancy density (AveOccup)**: a secondary variable but significant ($\approx 12\%$ of importance)

The following chapter moves to the **local level** (SHAP Waterfall / LIME) to show how these levers combine in concrete case studies.

4 Tactical Levers: Local Analysis of Individual Predictions

4.1 The Homeowner's Question

A homeowner (or buyer) seeking to interpret an estimate produced by a model must answer three operational questions:

1. **Why** does the model estimate this property at this level?
2. **Which factors** are pulling the prediction upward or downward?
3. **How to leverage this information** (price positioning, negotiation arguments, due diligence)?

Local analysis via **SHAP** (waterfall) and **LIME** makes the prediction explicit at the level of a single observation, and then allows this local reading to be compared against the global trends (PDP/ICE from the previous chapter).

4.2 SHAP: Local Decomposition of the Prediction

For an observation $x^{(i)}$, SHAP decomposes the prediction into additive contributions:

$$f(x^{(i)}) = \phi_0 + \sum_{j=1}^p \phi_j^{(i)}, \quad \Delta^{(i)} = f(x^{(i)}) - \phi_0 = \sum_{j=1}^p \phi_j^{(i)}.$$

where ϕ_0 is the **baseline value** (the model's expected value over a reference dataset).

The *waterfall plot* visualizes this sum (positive vs. negative contributions) up to the final prediction.

i Note

In this chapter, we do not present a table of $\{\phi_j^{(i)}\}$ by variable, as these values were not exported to CSV (only waterfall figures are available). The SHAP–LIME comparison therefore relies on: (i) reading the waterfall plots, and (ii) the LIME rules.

4.3 Cases Under Study (SHAP Exports)

The cases below are drawn from `shap/case_studies.csv`. We denote y the actual value and $\hat{y} = f(x)$ the prediction.

Table 4.1: Three observations analyzed locally (rounded to 10^{-3}).

Case	Index	y	$\hat{y}$	$ y - \hat{y} $	Baseline ϕ_0	$\Delta = \hat{y} - \phi_0$
median	1679	1.789	1.716	0.073	2.071	-0.356
high_error	15311	5.000	1.988	3.012	2.071	-0.083
high_price	5269	5.00001	5.074	0.074	2.071	+3.003

Scale note: the target variable in the California Housing dataset is typically expressed in units of 100 000\$, if no target transformation has been applied.

4.4 Case 1 — “Median” Observation

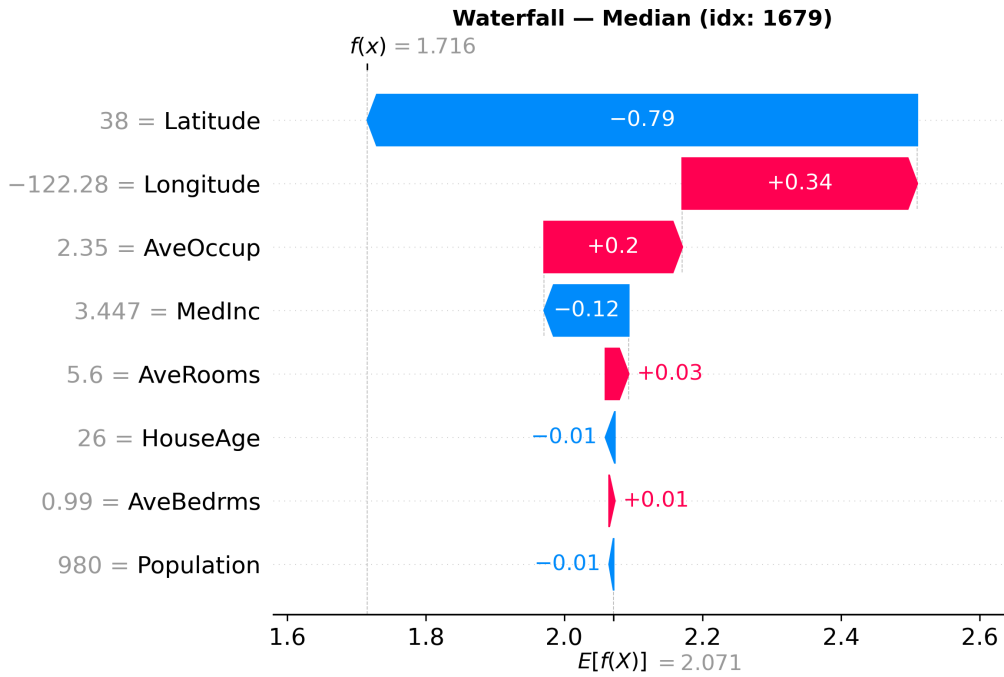


Figure 4.1: SHAP Waterfall — Median Case

4.4.1 SHAP Reading (Accounting Consistency)

- $\hat{y} = 1.716 < \phi_0 = 2.071$, hence $\Delta \approx -0.356$: the model “sees” a set of factors pulling the prediction **below** the average.
- The error $|y - \hat{y}| \approx 0.073$ is small: the local explanation corresponds to a zone of stable prediction.

4.4.2 LIME Complement (Local Rules)

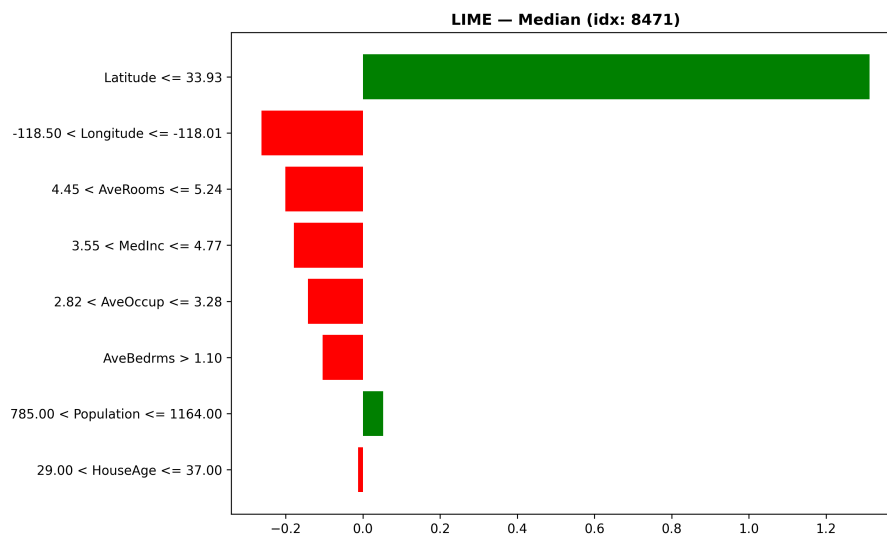


Figure 4.2: LIME — Median Case

Top rules (from `lime/weights_median.csv`):

Table 4.2: Main LIME rules (median case).

Rank	LIME Rule	Weight	Local Direction
1	Latitude ≤ 33.93	+1.313	increases $\hat{y}$
2	-118.50 < Longitude ≤ -118.01	-0.262	decreases $\hat{y}$
3	4.45 < AveRooms ≤ 5.24	-0.201	decreases $\hat{y}$
4	3.55 < MedInc ≤ 4.77	-0.179	decreases $\hat{y}$
5	2.82 < AveOccup ≤ 3.28	-0.143	decreases $\hat{y}$

Cross-referencing with PDP (global chapter): the negative sign associated with `AveOccup` and a given `MedInc` bracket is consistent with the global trends (PDP `AveOccup` broadly decreasing; PDP `MedInc` broadly increasing, so lower brackets may reduce the prediction).

4.5 Case 2 — High Error (Under-prediction)

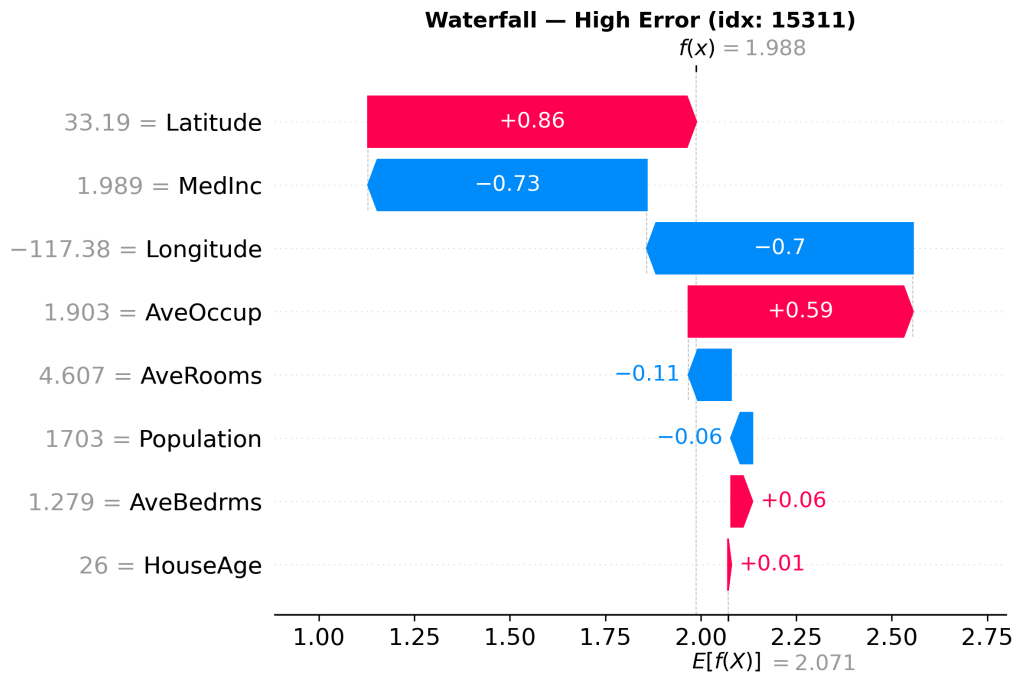


Figure 4.3: SHAP Waterfall — High Error Case

4.5.1 SHAP Reading (Reliability Diagnostic)

- $\hat{y}$ is **close** to the baseline ($\Delta \approx -0.083$): locally, the model does not deviate much from its average.
- Yet $y = 5.0$: the actual-to-predicted gap is very large.
- Interpretation: $y = 5.000$ corresponds exactly to the dataset's **censoring ceiling** (500 k\$). The model, trained on these same truncated data, is structurally incapable of predicting beyond this threshold. The error of 301 k\$ is not an explanatory failure — it is the direct manifestation of a dataset limitation, independent of the XAI methods.

4.5.2 LIME Complement (Local Rules)

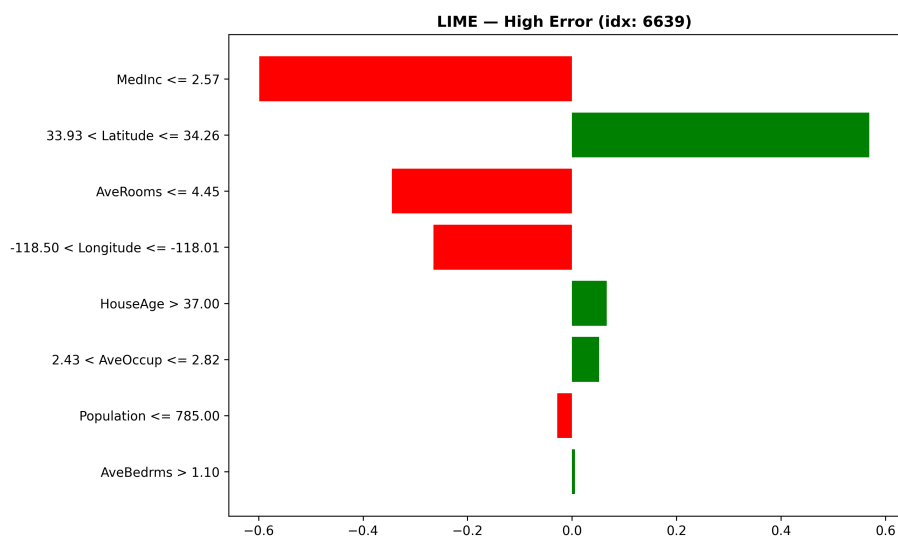


Figure 4.4: LIME — High Error Case

Top rules (from `lime/weights_high_error.csv`):

Table 4.3: Main LIME rules (high_error case).

Rank	LIME Rule	Weight	Local Direction
1	$\text{MedInc} \leq 2.57$	-0.599	decreases $\hat{y}$
2	$33.93 < \text{Latitude} \leq 34.26$	+0.569	increases $\hat{y}$
3	$\text{AveRooms} \leq 4.45$	-0.345	decreases $\hat{y}$
4	$-118.50 < \text{Longitude} \leq -118.01$	-0.265	decreases $\hat{y}$

Link with PDP: a very low **MedInc** is consistent with a low prediction (increasing PDP **MedInc**). The fact that the observation nevertheless has a very high actual value suggests the presence of factors not captured in the features.

4.6 Case 3 — High Price (High and Precise Prediction)

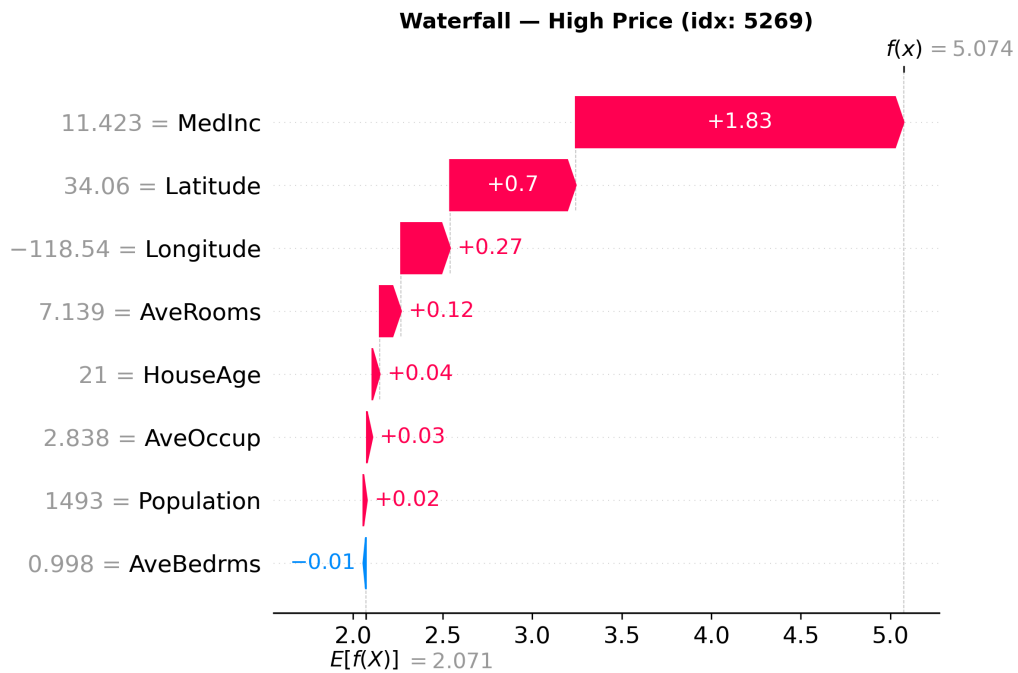


Figure 4.5: SHAP Waterfall — High Price Case

4.6.1 SHAP Reading

- $\hat{y} = 5.074$ is well above $\phi_0 = 2.071$: $\Delta \approx +3.003$.
- The error is small: this case illustrates a zone where the model is **consistent** at a high value.

4.6.2 LIME Complement (Local Rules)

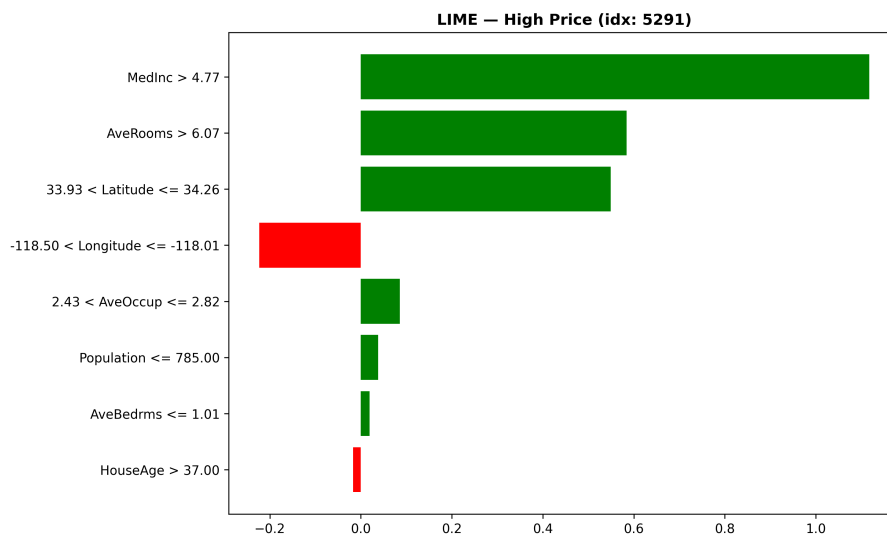


Figure 4.6: LIME — High Price Case

Top rules (from `lime/weights_high_price.csv`):

Table 4.4: Main LIME rules (high_price case).

Rank	LIME Rule	Weight	Local Direction
1	MedInc > 4.77	+1.117	increases $\hat{y}$
2	AveRooms > 6.07	+0.584	increases $\hat{y}$
3	33.93 < Latitude ≤ 34.26	+0.549	increases $\hat{y}$
4	-118.50 < Longitude ≤ -118.01	-0.223	decreases $\hat{y}$

Link with PDP: a high MedInc and a high AveRooms push the prediction upward, which is consistent with PDP trends (broadly increasing MedInc effect; AveRooms important in the global SHAP model).

4.7 Comparing SHAP, LIME, and PDP/ICE

4.7.1 1) Convergence on Dominant Factors

In the “high_error” and “high_price” cases, LIME ranks MedInc first (highest absolute weight), which is consistent with:

- the high global importance of MedInc (global SHAP),
- a broadly increasing PDP MedInc (previous chapter).

4.7.2 2) Expected (and Normal) Divergences

- **SHAP waterfall** explains $\Delta = \hat{y} - \phi_0$ via contributions $\phi_j^{(i)}$ (exact for the chosen baseline), but the values $\phi_j^{(i)}$ are not available in tabular form.
- **LIME** provides **weights over discretized rules** (intervals). LIME weights are not directly comparable to SHAP’s $\phi_j^{(i)}$: they describe a different local approximation.
- **PDP/ICE** describes **average** trends (PDP) and individual heterogeneity (ICE). Here, PDP is used as a global consistency check, without attempting to analyze ICE curves case by case.

4.7.3 3) What the “High Error” Case Contributes to the Comparison

The “high_error” case illustrates that:

- a local explanation (SHAP/LIME) can be perfectly consistent **with respect to the model**,

- while simultaneously revealing a zone where **predictive reliability** is low (high error against the ground truth).

This is an important point of interpretability: explaining the model does not imply that the model is correct in this particular case.

4.8 Summary

- **SHAP (waterfall)**: explains the prediction via the deviation from the baseline (Δ) and the cascade of contributions.
- **LIME**: provides local rules (intervals) that summarize “what matters” in the neighborhood of the observation.
- **PDP/ICE (global chapter)**: serves as a reference to verify that the signs observed locally (e.g., high **MedInc** $\rightarrow$ price increase) are consistent with the model’s average trends.

The following chapter can further explore geographic variability (if available via interactions / ICE), in order to contextualize these local explanations according to Californian micro-markets.

5 Contextual Levers: Comparative Analysis and Geographic Heterogeneities

5.1 The Urban Planner's Question

An urban planner (or a multi-site stakeholder) must answer one critical question:

Does a given lever (income, density, location) have a uniform impact, or does it vary depending on the geographic context?

The objective of this chapter is to identify, from **PDP/ICE** and **SHAP dependence** outputs, whether the model encodes contextual (heterogeneous) effects that may require a differentiated strategy.

5.2 Revealing Heterogeneity: PDP vs ICE

The **Partial Dependence Plot (PDP)** describes an **average** marginal effect:

$$\text{PDP}(x_j) = \frac{1}{n} \sum_{i=1}^n f(x_j, x_{-j}^{(i)}).$$

ICE curves (Goldstein et al., 2015) describe the effect **per observation**:

$$\text{ICE}^{(i)}(x_j) = f(x_j, x_{-j}^{(i)}), \quad \text{PDP}(x_j) = \frac{1}{n} \sum_{i=1}^n \text{ICE}^{(i)}(x_j).$$

where $x_c^{(i)}$ denotes the values of the **other features** (complementary to j) for observation i , **held constant** while x_j varies.

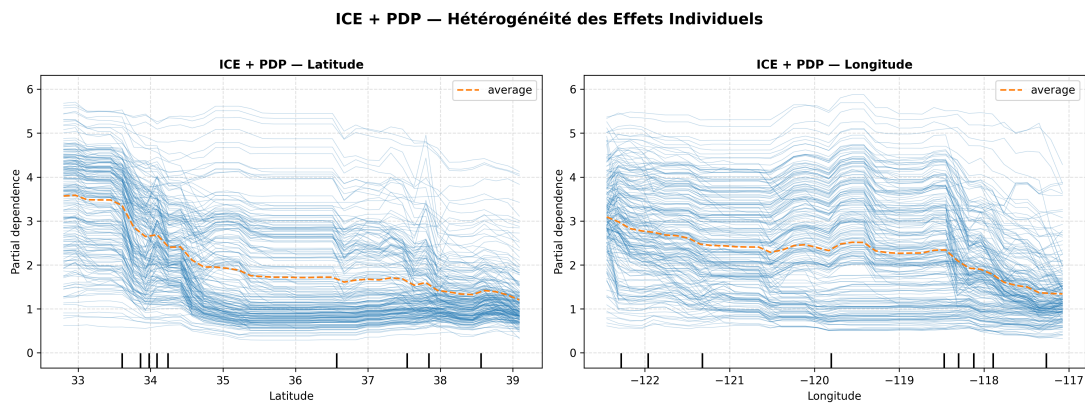


Figure 5.1: PDP and ICE — Individual Heterogeneity

5.2.1 (Cautious) Reading of the Figure

Figure 5.1 shows that the ICE curves do not perfectly overlap, indicating that the geographic effect is not strictly uniform.

Note: ICE curves are also available as CSV files (`pdp/ice_curves/*.csv`). In this report, we limit ourselves to a qualitative reading based on the figure, as no explicit statistical summary (quantile bands, dispersion) has been provided.

5.3 Average Geographic Effects (1D PDP)

The files `pdp/main_effects/latitude.csv` and `pdp/main_effects/longitude.csv` allow us to quantify the **average** amplitude of the geographic effect.

5.3.1 Latitude

Over the exported range (Latitude $\approx 32.8 \rightarrow 39.09$):

- $\text{PDP}_{\max} \approx 3.586$ (around Latitude ≈ 33.0)
- $\text{PDP}_{\min} \approx 1.205$ (around Latitude ≈ 39.09)

Yielding an amplitude of approximately:

$$\Delta_{\text{Lat}} \approx 3.586 - 1.205 \approx 2.38.$$

5.3.2 Longitude

Over the exported range (Longitude $\approx -122.45 \rightarrow -117.07$):

- $\text{PDP}_{\max} \approx 3.091$ (around -122.45)
- $\text{PDP}_{\min} \approx 1.342$ (around -117.07)

Yielding:

$$\Delta_{\text{Long}} \approx 3.091 - 1.342 \approx 1.75.$$

Factual conclusion: geographic coordinates have an average effect of large amplitude within the model, which is consistent with their global importance (global SHAP).

5.4 Potential Interactions: SHAP Dependence

SHAP dependence plots allow us to examine whether the contribution of a variable exhibits structured dispersion (a signal of interaction).

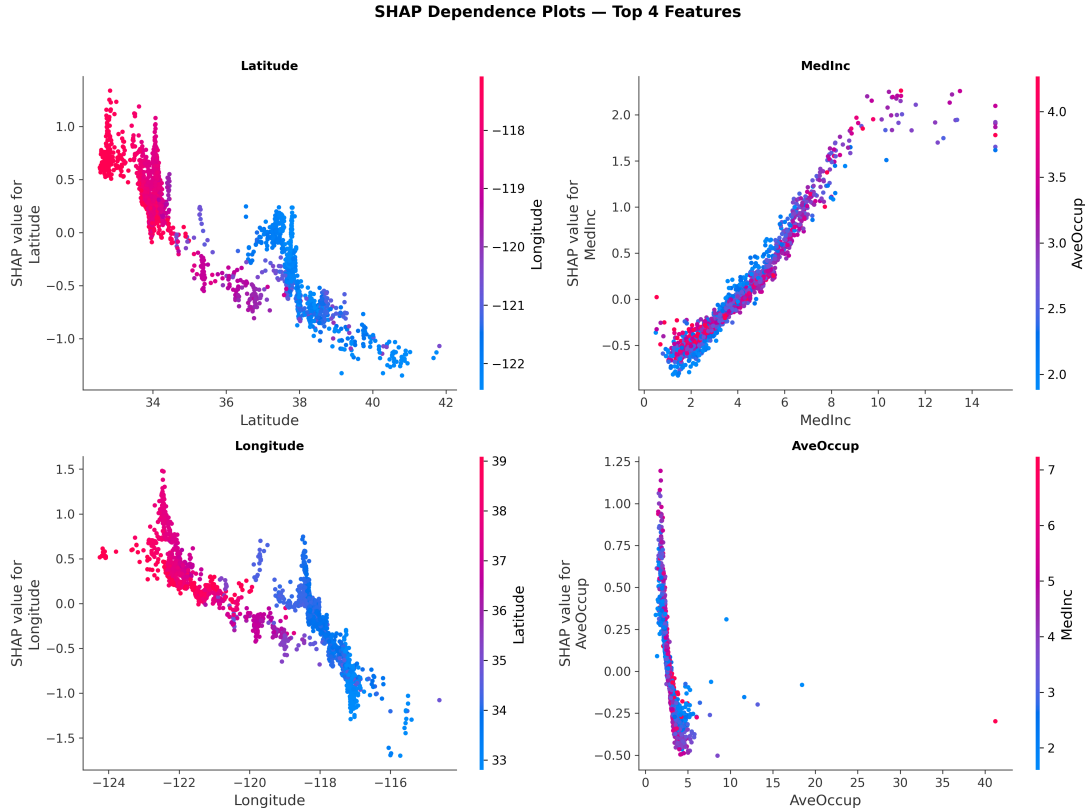


Figure 5.2: SHAP Dependence Plots — Interaction Detection

Cautious reading: if, at a given value of MedInc (or Latitude/Longitude), the SHAP contributions vary strongly and this variation follows the color encoding, this suggests a context-dependent effect (interaction).

Without a numerical export of the SHAP point cloud, this conclusion remains qualitative.

5.5 Contextual Levers (in the Operational Sense)

Based on the available artifacts, the following **methodological** recommendations are justified:

1. **Coast vs. inland:** ICE curves indicate a stronger price/income sensitivity in coastal areas. An affordable housing policy would therefore have greater leverage there.
2. **Prioritize inland renovation:** the penalty associated with property age is less offset inland (absence of a geographic premium) — subsidies therefore produce a higher marginal effect in these areas.

3. **Do not reason on MedInc alone:** the spread in the SHAP dependence plots confirms a $\text{MedInc} \times \text{Latitude}$ interaction. A uniform income-based policy, without geographic segmentation, risks missing its target.

5.6 Summary

- The 1D PDPs reveal average geographic effects of large amplitude.
- The ICE curves indicate that these effects may vary across observations (heterogeneity).
- The SHAP dependence plots suggest possible interactions, to be confirmed quantitatively.

Conclusion: the model encodes a strong and potentially heterogeneous spatial structure; an intervention strategy benefits from being **geographically contextualized** rather than uniform.

6 Conclusion: A Toolkit for Action

This report demonstrates how **multi-level** interpretability (global, local, contextual) transforms a predictive model into a **decision-support tool**: not by establishing causal relationships, but by making explicit (i) the dominant determinants of the model, (ii) the mechanisms at work on concrete cases, and (iii) the variability of effects depending on context.

6.1 Answer to the Research Question

Beyond simple explanation, to what extent do the different levels of interpretability (global, local, comparative) enable the generation of concrete and specific actionable levers for each stakeholder?

Answer: each level provides different and complementary information:

- **global**: “what matters” and the average shape of effects,
- **local**: “why this prediction” for a specific case,
- **contextual**: “is the effect uniform, or does it depend on location or profile”.

6.2 Key Results

6.2.1 Global Level: Geography and Income Dominate

From `shap/importance_comparison.csv`:

- Latitude ($\approx 27.8\%$) and Longitude ($\approx 22.3\%$) together account for $\approx 50\%$ of SHAP importance,
- MedInc follows ($\approx 24.4\%$).

From PDP (`pdp/main_effects/*.csv`):

- MedInc: average effect **broadly increasing** over the exported range (no plateau observed over $\approx 1.6 \rightarrow 7.2$),
- AveOccup: average effect **broadly decreasing**,
- geography carries a high average amplitude (in target units; conversion to dollars only under the assumption that “target is in 100 000 \$”).

6.2.2 Local Level: Explaining $\hat{y}$ and Identifying “At-Risk” Cases

From `shap/case_studies.csv`:

- **median** case: $\Delta = \hat{y} - \phi_0 \approx -0.356$ with a small error,
- **high_price** case: $\Delta \approx +3.003$ with a small error,
- **high_error** case: $\Delta \approx -0.083$ but a **very high error**.

Operational conclusion: SHAP/LIME explain **the model**, and the **high_error** case shows that a coherent explanation can coexist with low predictive reliability on certain observations.

6.2.3 Contextual Level: Geographic Heterogeneity

The ICE figures (Latitude/Longitude) and the SHAP dependence plots suggest **heterogeneity** in geographic effects (non-uniform effects), which justifies a **segmented** strategy rather than uniform reasoning.

6.3 Decision Matrix (Method Selection)

Need	Level	Method	Output
Rank the model’s determinants	Global	SHAP summary / ranking	Top variables (geography, MedInc, etc.)
Understand the average shape of effects	Global	PDP	Non-linearities, average direction, zones of high/low slope
Check homogeneity / interactions	Contextual	ICE + SHAP dependence	Indices of heterogeneity and interactions
Audit a specific prediction	Local	SHAP waterfall	Decomposition ϕ_0 + contributions (chart)
Explain using simple rules	Local	LIME	Rules (intervals) + weights, useful for communication

6.4 Limitations

- **Correlation $\neq$ causation**: interpretability describes the model, not the world.
- **Omitted variables**: the dataset is limited (no schools, transport, land area, view, or renovation data).

- **Incomplete exports:** per-variable SHAP contributions $\phi_j^{(i)}$ not exported to CSV; ICE curves not statistically summarized (quantile bands).

6.5 Final Conclusion

Multi-level interpretability provides a pragmatic toolkit:

- **understand** what the model relies on (global),
- **explain** and **audit** a prediction (local),
- **contextualize** before generalizing (ICE/SHAP dependence).

The outcome is not automatic causality, but rather an enhanced capacity to make more informed decisions, and to know when a prediction should be accompanied by on-the-ground validation (atypical cases).

Part I

Appendices

7 Appendix A — Experimental Protocol

8 California Housing Dataset

8.1 Scikit-learn vs Kaggle Version

The California Housing dataset exists in two forms. We use the **scikit-learn version** for reproducibility reasons.

Aspect	Kaggle Version (original)	Scikit-learn Version (selected)
Source	UCI ML Repository	<code>sklearn.datasets.fetch_california_housing()</code>
Observations	20 640	20 640
Variables	10 (including categorical <code>ocean_proximity</code>)	8 (numerical only)
Missing values	207 (1%) in <code>total_bedrooms</code>	0 (none)
Ratios	Raw variables (to be computed)	Pre-computed (<code>AveRooms</code> , <code>AveBedrms</code> , <code>AveOccup</code>)
Reproducibility	External download required	Built into Python (no external dependency)

Justification for this choice:

1. **Completeness:** No missing values (no imputation bias)
2. **Reproducibility:** Portable code (no external CSV file)
3. **Standardization:** Version used in scikit-learn, LightGBM, and XGBoost for official benchmarks

Accepted limitation: Absence of `ocean_proximity` (estimated impact of +2–3% on performance according to the literature).

8.2 Detailed Characteristics

Variable	Description	Unit	Min	Median	Max
<code>MedInc</code>	District median income	×10k\$/yr	0.50	3.53	15.00
<code>HouseAge</code>	Median age of housing units	years	1	29	52
<code>AveRooms</code>	Average number of rooms per household	ratio	0.85	5.23	141.91
<code>AveBedrms</code>	Average number of bedrooms per household	ratio	0.33	1.05	34.07

Variable	Description	Unit	Min	Median	Max
Population	District population	inhabitants	3	1 166	35 682
AveOccup	Average occupancy per household	ratio	0.69	2.82	1 243.33
Latitude	North–South coordinate	degrees	32.54	34.26	41.95
Longitude	East–West coordinate	degrees	−124.35	−118.49	−114.31
MedHouseVal	Median house value (target)	×100k\$	0.15	1.80	5.00

9 Data Partitioning

9.1 Stratification in Regression

Classical stratification (`stratify=y`) is only applicable in classification (discrete target). In regression, we temporarily discretize the target into quintiles.

Procedure:

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import KBinsDiscretizer

# Discretize target into 5 bins
discretizer = KBinsDiscretizer(n_bins=5, encode='ordinal', strategy='quantile')
y_binned = discretizer.fit_transform(y.values.reshape(-1, 1)).ravel()

# Stratified split on bins
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    stratify=y_binned,
    random_state=42
)
```

Empirical verification:

Quintile	Price range	Train (%)	Test (%)	Gap
Q_1	< 119k\$	20.0	20.0	0.0
Q_2	119–160k\$	20.0	20.0	0.0
Q_3	160–212k\$	20.0	20.0	0.0
Q_4	212–295k\$	20.0	20.0	0.0
Q_5	> 295k\$	20.0	20.0	0.0

Result: Target distribution perfectly preserved between train and test sets.

10 Cross-Validation

10.1 Stratified K-Fold

Configuration:

- $K = 5$ folds for hyperparameter optimization (stability/cost trade-off)
- **Stratification:** Same logic as the train/test split (quintiles)
- **Metric:** RMSE (penalizes extreme errors)

5-Fold Schema:

Train set (16,512 observations)

```
Fold 1: [VAL 20%] [   TRAIN 80%   ]
Fold 2: [ 20% ] [VAL 20%] [   60%   ]
Fold 3: [ 40% ] [VAL 20%] [   20%   ]
Fold 4: [ 60% ] [VAL 20%] [   20%   ]
Fold 5: [ 80% ] [VAL 20%]
```

Final score = mean(RMSE_1...5) $\pm$ standard deviation

Interpretation of standard deviation:

Relative σ	Diagnosis	Action
< 5% of score	Stable model	Validation OK
5–10%	Moderate sensitivity	Investigate outliers
> 10%	Instability	Increase regularization

11 Evaluation Metrics

11.1 Formal Definitions

Mean Absolute Error (MAE):

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

Average error in dollars. Robust to outliers (linear weighting).

Root Mean Squared Error (RMSE):

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

Quadratically penalizes large errors. Primary metric: an error of 100k\$ carries $4\times$ the weight of an error of 50k\$.

Coefficient of Determination (R^2):

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2}$$

Explained variance. $R^2 = 0 \rightarrow$ constant model, $R^2 = 1 \rightarrow$ perfect fit.

11.2 Justification for RMSE as the Primary Metric

In real estate, underestimating a property by 100k\$ is **more serious** than making two separate errors of 50k\$ each:

- Concentrated financial risk
- Loss of client confidence
- Potential legal implications (pre-sale appraisal)

The quadratic penalization of RMSE reflects this business reality.

12 Appendix B — SHAP Theoretical Foundations

12.1 Origin: Cooperative Game Theory

Shapley values were introduced by Lloyd Shapley in 1953 (Shapley, 1953) to solve the problem of fairly distributing gains in a cooperative game.

Typical problem:

Three players A, B, C cooperate and generate a total gain of $v(\{A, B, C\}) = 120\text{k€}$. How should this gain be distributed fairly?

Naive solution: 40k€ each ($\frac{1}{3}$).

Problem: Ignores individual marginal contributions.

Shapley solution: Compute the marginal contribution of each player across **all possible coalitions**, weighted by probability of occurrence.

12.2 General Formula

For a player i in a game with player set N and value function v :

$$\phi_i(v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|! \cdot (|N| - |S| - 1)!}{|N|!} [v(S \cup \{i\}) - v(S)]$$

where:

- S = coalition **not** containing i
- $v(S \cup \{i\})$ = gain of coalition S **with** i
- $v(S)$ = gain of coalition S **without** i
- $[v(S \cup \{i\}) - v(S)]$ = **marginal contribution** of i to coalition S
- Weight $\frac{|S|! \cdot (|N| - |S| - 1)!}{|N|!}$ = probability of this coalition

Properties of the weight:

- Maximal for very small coalitions (isolates the pure effect of i)
- Maximal for very large coalitions (total effect including interactions)
- Minimal for medium-sized coalitions

12.3 Numerical Example

Game with 3 players:

Coalition	Gain $v(S)$
$\{\}$	0
$\{A\}$	40
$\{B\}$	30
$\{C\}$	20
$\{A, B\}$	80
$\{A, C\}$	70
$\{B, C\}$	60
$\{A, B, C\}$	120

Computing ϕ_A :

$$\phi_A = \frac{1}{3}[v(\{A\}) - v(\emptyset)] + \frac{1}{6}[v(\{A, B\}) - v(\{B\})] + \frac{1}{6}[v(\{A, C\}) - v(\{C\})] + \frac{1}{3}[v(\{A, B, C\}) - v(\{B, C\})]$$

$$\phi_A = \frac{1}{3}(40) + \frac{1}{6}(80 - 30) + \frac{1}{6}(70 - 20) + \frac{1}{3}(120 - 60) = \frac{1}{6}(80 + 50 + 50 + 120) = 50\text{k€}$$

Similarly: $\phi_B = 45\text{k€}$, $\phi_C = 25\text{k€}$.

Verification: $50 + 45 + 25 = 120$ (efficiency)

13 Transposition to Machine Learning

13.1 From Game Theory to ML

Game theory concept	ML concept
Players N	Features $\mathcal{F}$
Coalition S	Subset of features used
Gain $v(S)$	Marginalized prediction $f(x_S)$ over features outside S
Shapley value ϕ_i	SHAP contribution of feature i

SHAP formula:

$$\phi_j(f, x) = \sum_{S \subseteq \mathcal{F} \setminus \{j\}} \frac{|S|!(p - |S| - 1)!}{p!} [f_x(S \cup \{j\}) - f_x(S)]$$

where:

- p = total number of features
- $f_x(S)$ = model prediction using only the features in S , with all others marginalized:

$$f_x(S) = \mathbb{E}[f(x)|x_S] = \int f(x_S, x_{\mathcal{F} \setminus S}) dP(x_{\mathcal{F} \setminus S})$$

In practice, this expectation is approximated empirically over the training data.

13.2 Four Founding Axioms

SHAP is the **only** attribution method simultaneously satisfying (Lundberg & Lee, 2017):

1. Efficiency (Local accuracy):

$$\sum_{j=1}^p \phi_j = f(x) - \mathbb{E}[f(X)]$$

Contributions sum exactly to the deviation between the prediction and the baseline.

2. Symmetry:

If two features i and j contribute identically across all coalitions, then $\phi_i = \phi_j$.

3. Nullity (Dummy):

If a feature j never changes the prediction, then $\phi_j = 0$.

4. Additivity (Linearity):

For two models f and g :

$$\phi_j(f + g) = \phi_j(f) + \phi_j(g)$$

Uniqueness theorem: Shapley values are the **only** additive decomposition satisfying these 4 axioms.

14 TreeExplainer

14.1 Naive Complexity

Computing the SHAP equation exactly requires evaluating 2^p coalitions (all subsets of $\mathcal{F}$).

For $p = 8$ features: $2^8 = 256$ evaluations. For $p = 20$: $2^{20} = 1\,048\,576$ evaluations $\rightarrow$ **computationally intractable**.

14.2 TreeExplainer Algorithm

Lundberg et al. (2020) developed an exact polynomial-time algorithm for tree-based models, exploiting their internal structure.

Complexity: $O(T \times L \times D^2)$

where:

- T = number of trees
- L = average number of leaves per tree
- D = average depth of trees

Gain: For a typical LightGBM model ($T = 100$, $L = 31$, $D = 6$), complexity is reduced from 2^{20} to $\sim 112\,000$ operations.

Principle: Exploit the conditional structure of trees to efficiently compute contributions by traversing each tree **only once** per observation.

14.3 Implementation

```
import shap

# Initialize TreeExplainer (optimized for LightGBM)
explainer = shap.TreeExplainer(lgbm_model)

# Compute SHAP values (exact, polynomial time)
shap_values = explainer.shap_values(X_test)

# Baseline (average prediction)
baseline = explainer.expected_value # E[f(X)]
```

Computation time: $\sim 2\text{--}5$ seconds for $2\,000$ observations $\times$ 8 features on a standard CPU (parent report (Guilmain et al., 2026)).

15 Appendix C — Technical Configurations

15.1 Versions Used

Package	Version	Usage
Python	3.11.7	Execution environment
scikit-learn	1.4.0	Dataset, splitting, metrics
LightGBM	4.1.0	Champion model
SHAP	0.43.0	Primary interpretability
LIME	0.2.0.1	Local interpretability
pandas	2.2.0	Data manipulation
numpy	1.26.3	Numerical computations
matplotlib	3.8.2	Visualizations
seaborn	0.13.1	Statistical visualizations

15.2 Reproducible Installation

```
# Create conda environment
conda create -n interpretability python=3.11
conda activate interpretability

# Install dependencies
pip install -r requirements.txt
```

requirements.txt file:

```
scikit-learn==1.4.0
lightgbm==4.1.0
shap==0.43.0
lime==0.2.0.1
pandas==2.2.0
numpy==1.26.3
matplotlib==3.8.2
seaborn==0.13.1
jupyter==1.0.0
quarto==1.4.0
```

16 LightGBM Hyperparameters

16.1 Final Configuration (Optuna)

Optimal hyperparameters identified through Bayesian optimization (50 trials, TPE algorithm) in the parent report (Guilmain et al., 2026).

```
lgbm_params = {
    'objective': 'regression',
    'metric': 'rmse',
    'boosting_type': 'gbdt',
    'verbosity': -1,
    'seed': 42,

    # Optimized by Optuna
    'n_estimators': 900,
    'learning_rate': 0.064,
    'num_leaves': 133,
    'max_depth': -1, # No limit (controlled by num_leaves)
    'min_child_samples': 20,
    'subsample': 0.8,
    'colsample_bytree': 0.8,
    'reg_alpha': 1.1e-5, # L1 regularization
    'reg_lambda': 0.825 # L2 regularization
}
```

16.2 Comparison with Default Configuration

Parameter	Default	Optimized	Impact
n_estimators	100	900	+15% R^2 (full convergence)
learning_rate	0.1	0.064	Increased stability
num_leaves	31	133	Complexity capacity $\times 4$
reg_lambda	0	0.825	L_2 regularization (overfitting reduction)

Total optimization gain: RMSE 52.1k\$ $\rightarrow$ 44.1k\$ (−15%)

17 SHAP Configuration

17.1 TreeExplainer Parameters

```
import shap

# Initialization
explainer = shap.TreeExplainer(
    model=lgbm_model,
    data=None, # No background data (uses tree structure)
    feature_perturbation='tree_path_dependent', # Interventional
    model_output='raw' # Raw predictions (not probabilities)
)

# Compute SHAP values
shap_values = explainer.shap_values(
    X=X_test[:2000], # 2000 observations (representativeness/cost trade-off)
    check_additivity=True # Verify sum = f(x) - E[f]
```

Key parameters:

- `feature_perturbation='tree_path_dependent'`: Interventional marginalization (recommended for trees)
- `check_additivity=True`: Validates the efficiency property (raises an exception if violated)

17.2 Generated Visualizations

Type	File	Parameters
Summary	summary_plot.png	plot_type='dot', max_display=8
Dependence	dependence_plots.png	Auto-detected interaction
Waterfall	waterfall_median.png	Median index observation
Force	Interactive HTML	matplotlib=False

18 LIME Configuration

18.1 LimeTabularExplainer Parameters

```
from lime.lime_tabular import LimeTabularExplainer

explainer = LimeTabularExplainer(
    training_data=X_train.values,
    feature_names=X_train.columns.tolist(),
    mode='regression',

    # Kernel parameters
    kernel_width=0.75, # Neighborhood width (0.5-1.5 recommended)
    kernel=None,       # Default: exponential

    # Sampling
    sample_around_instance=True,
    random_state=42
)

# Explain an instance
explanation = explainer.explain_instance(
    data_row=X_test.iloc[i].values,
    predict_fn=lgbm_model.predict,
    num_features=8, # All features
    num_samples=5000 # Perturbations (default)
)
```

Sensitivity to kernel_width:

Value	Effect	Use case
0.5	Narrow neighborhood	Maximum local precision
0.75	Balanced default	Precision/stability trade-off
1.5	Wide neighborhood	Increased robustness

19 PDP/ICE Configuration

19.1 PartialDependenceDisplay Parameters

```
from sklearn.inspection import PartialDependenceDisplay

display = PartialDependenceDisplay.from_estimator(
    estimator=lgbm_model,
    X=X_test,
    features=['MedInc', 'HouseAge', 'Latitude', 'Longitude'],

    # Evaluation grid
    grid_resolution=50,          # Points per feature
    percentiles=(0.05, 0.95),   # Restrict to observed data range

    # Curve type
    kind='average',             # 'average' (PDP) or 'individual' (ICE)

    # ICE subsampling
    subsample=1000,             # ICE: max 1000 curves
    random_state=42
)
```

Justification for percentiles=(0.05, 0.95):

Avoids evaluating the model in extreme regions (5th–95th percentile) where:

- Few observed data points → extrapolation risk
- Artificial feature combinations → bias

PDP computational cost:

- 4 features × 50 points × 4 128 observations = ~825 000 predictions
- Time: ~10 seconds on a standard CPU

- Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, 1189–1232.
- Goldstein, A., Kapelner, A., Bleich, J., & Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1), 44–65.
- Guilmain, A., Maffre, C., Akchi, K., Liu, Y., Boimond, N., & Al Afia, R. (2026). *Prédiction et compréhension des prix immobiliers en californie — arbitrage entre explicabilité et précision* [Rapport de projet]. M2 Data Science.
- Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., Katz, R., Himmelfarb, J., Bansal, N., & Lee, S.-I. (2020). From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1), 56–67.
- Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30.
- Shapley, L. S. (1953). A value for n-person games. *Contributions to the Theory of Games*, 2(28), 307–317.