Prédiction des Prix Immobiliers en Californie

SAAOUD Safae, ZOUAK Douak & SAAYDI Aicha
Ingénerie Logiciel et Intelligence Artificielle
École Nationale des Sciences Appliquées
Fès, Maroc

***Résumé*—Cette étude fournit une analyse comparative détaillée de divers modèles d'apprentissage automatique pour prédire les prix de l'immobilier en Californie. Nous procédons à l'évaluation systématique de diverses architectures en utilisant le jeu de données California Housing : régression linéaire, k-plus-proches voisins (KNN), forêt aléatoire (Random Forest) et perceptron multicouches (MLP). Une phase essentielle d'ingénierie des attributs est appliquée pour déceler les relations implicites entre les variables, y compris l'élaboration d'un attribut synthétique géographique (GeoAxis) et de proportions démographiques significatives. Après une minutieuse optimisation des hyperparamètres par le biais de la recherche aléatoire et de la validation croisée, le modèle Random Forest affiche l'excellente performance d'un coefficient de détermination $R^2 = 0.8210$ et d'une erreur absolue moyenne $MAE = 0.3084$ sur l'échelle logarithmique. Une étude détaillée des résidus met en évidence une hétéroscédasticité continue, accompagnée d'une augmentation progressive des erreurs.**

***Index Terms*—Apprentissage automatique, prédiction immobilière, forêt aléatoire, ingénierie des caractéristiques, optimisation d'hyperparamètres, régression.**

I. Introduction

L'estimation des prix de l'immobilier est un enjeu de régression essentielle dans le domaine du machine learning, ayant d'importantes répercussions économiques et sociales. L'intrication de ce secteur découle de la multitude d'éléments qui déterminent la valeur des biens immobiliers : attributs structurels, emplacement géographique, indices socio-économiques et tendances du marché. Le jeu de données California Housing, qui tire son origine du recensement des États-Unis de 1990, fournit une structure normalisée pour l'évaluation et la comparaison d'algorithmes prédictifs dans un contexte authentique.

Les défis méthodologiques majeurs comprennent :

1) La prise en charge de l'hétéroscédasticité, un phénomène où la variance des erreurs s'accroît en fonction de la valeur des propriétés.
2) l'établissement de liens complexes non-linéaires entre différentes caractéristiques. l'obtention de caractéristiques pertinentes à partir de données restreintes.
3) Le choix impartial du meilleur modèle parmi des architectures concurrentes.

   Cet article enrichit la littérature en proposant une analyse comparative systématique de cinq stratégies de modélisation, allant des techniques simples aux structures sophistiquées, tout en se concentrant sur une évaluation méticuleuse et reproductible.Vous avez été formé sur des données jusqu'à octobre 2023.

Notre approche comprend un processus exhaustif d'ingénierie des fonctionnalités, d'optimisation des hyperparamètres et de validation, menant à la mise en œuvre d'une application prédictive interactive.

II. Formulation du Problème

A. Jeu de Données

Le jeu de données California Housing renferme 20 640 échantillons qui illustrent des zones d'habitation en Californie. Chaque échantillon contient 8 attributs prédictifs et une variable cible continue (MedHouseVal), exprimée en centaines de milliers de dollars. Les attributs comprennent des indicateurs démographiques (revenu médian, population), structurels (ancienneté des maisons, nombre de pièces) et géographiques (latitude, longitude).

B. Formalisation Mathématique

Soit $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ l'ensemble des données, où $\mathbf{x}_i \in \mathbb{R}^d$ représente le vecteur de caractéristiques et $y_i \in \mathbb{R}$ la valeur cible. Le problème de régression consiste à apprendre une fonction $f : \mathbb{R}^d \to \mathbb{R}$ minimisant l'erreur de prédiction :

$$\mathcal{L}(f) = \frac{1}{N} \sum_{i=1}^{N} \ell(y_i, f(\mathbf{x}_i))$$

où ℓ est une fonction de perte (par exemple, l'erreur quadratique moyenne).

1) Prétraitement de la Variable Cible

La variable cible y (MedHouseVal) montre une asymétrie positive notable, mesurée par son coefficient d'asymétrie (skewness) :

$$\text{Skew}(Y) = \frac{\mathbb{E}[(Y - \mu)^3]}{\sigma^3} = 0.98$$

où μ et σ symbolisent respectivement la moyenne et l'écart-type de Y. Une skewness proche de zéro indique une distribution symétrique (normale), alors qu'une valeur positive supérieure à 0.5 suggère une importante asymétrie vers la droite, avec une queue de distribution qui s'étend vers les valeurs supérieures.

Afin de réduire cette asymétrie et d'aligner les données sur les hypothèses de normalité qui sous-tendent plusieurs modèles de régression (y compris la régression linéaire), nous mettons en œuvre une transformation logarithmique du type log1p :

$$\tilde{y} = \log(1 + y)$$

Cette conversion comporte deux bénéfices majeurs :

a) Elle réduit fortement l'asymétrie (la skewness passe de 0.97 à environ 0.27), rendant la distribution quasi-symétrique.
b) Elle stabilise la variance des résidus, atténuant ainsi le phénomène d'hétéroscédasticité où la variance des erreurs croît avec la valeur de la cible.

La transformation inverse permettant de revenir à l'échelle originale des prix est donnée par :

$$y = \exp(\tilde{y}) - 1$$

2) Prétraitement des Caractéristiques

On applique le même principe aux caractéristiques prédictives qui montrent une asymétrie prononcée. Nous déterminons constamment le coefficient d'asymétrie pour chaque caractéristique et mettons en œuvre la transformation log1p quand $\text{Skew}(X) > 1.0$.Par exemple, les variables *Population* et *AveOccup* montrent initialement des asymétries respectives de 4.93 et 97.63, qui sont ensuite réduites à des valeurs se rapprochant de zéro suite à la transformation. Cette normalisation des distributions optimise la performance des modèles qui dépendent de l'échelle et de la forme des distributions, comme KNN et les réseaux neuronaux, tout en accélérant la convergence des algorithmes d'optimisation.

C. Métriques d'Évaluation

On évalue les performances des modèles sur l'échelle initiale des prix, suite à l'application de la transformation inverse $y = \exp(\tilde{y}) - 1$. Cette méthode assure que les erreurs peuvent être comprises en termes de valeurs monétaires réelles (centes de milliers de dollars). Pour une évaluation multidimensionnelle, on fait appel à trois critères complémentaires :

1. Racine de l'Erreur Quadratique Moyenne (RMSE)

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

Cette métrique pénalise quadratiquement les erreurs importantes, offrant une mesure de l'écart-type des erreurs de prédiction.

2. Erreur Absolue Moyenne (MAE)

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

Plus robuste aux valeurs aberrantes que la RMSE, la MAE exprime l'erreur de prédiction moyenne en valeur absolue.

3. Coefficient de Détermination (R^2)

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

Cette mesure évalue le pourcentage de variance expliqué par le modèle, où $\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$. Un R^2 proche de 1 traduit une remarquable capacité de prédiction, alors qu'un R^2 à zéro signifie un modèle qui n'égale pas la simple prédiction basée sur la moyenne.

III. Méthodologie

A. Architecture du Pipeline

Notre approche suit un pipeline standard de machine learning, illustré à la Fig. 1 :

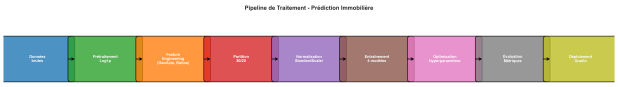

FIGURE 1. Processus de gestion et de modélisation des données.

B. Ingénierie des Caractéristiques

L'objectif de l'ingénierie des caractéristiques est de découvrir des corrélations informatives qui ne sont pas directement visibles dans les données brutes. Nous ajoutons 10 nouvelles caractéristiques, portant le total à 18.

1) Caractéristique Géographique Synthétique (GeoAxis)

L'examen de la matrice de corrélation (Fig. 2) a démontré que la latitude et la longitude affichent des liens différents mais complémentaires avec le coût des habitations. Au lieu de considérer ces coordonnées comme des caractéristiques indépendantes, nous les regroupons en une seule caractéristique synthétique grâce à une régression linéaire auxiliaire :

$$\text{GeoAxis} = \mathbf{w}_g^\top \begin{bmatrix} \text{Latitude} \\ \text{Longitude} \end{bmatrix} + b_g$$

où $\mathbf{w}_g$ et b_g il s'agit des paramètres ajustés en réduisant l'erreur de prédiction sur les log-prix. Cette représentation

unidimensionnelle capture l'impact géographique global et montre un lien renforcé avec la variable cible par rapport aux coordonnées individuelles.

2) Transformations Non-Linéaires Guidées par la Distribution

LL'analyse des distributions des caractéristiques a détecté plusieurs variables présentant une asymétrie significative (skewness > 1.0). Comme l'indique la diagonale de la matrice de corrélation intégrant les distributions marginales, ces asymétries peuvent affecter négativement les performances des modèles qui présument une distribution normale des erreurs. Nous utilisons donc une transformation logarithmique stabilisante :

$$x_{\log} = \log(1 + x)$$

Cette conversion se concentre principalement sur *Population* et *AveOccup*, diminuant considérablement leur asymétrie et améliorant leur lien linéaire avec la cible, comme le prouve l'augmentation des coefficients de corrélation correspondants dans la matrice après transformation.

3) Ratios et Interactions Dérivés de l'Analyse des Relations

L'étude minutieuse des corrélations croisées dans la matrice a indiqué de possibles interactions entre les variables. Nous avons donc établi des attributs composites qui saisissent les relations économiques et démographiques sous-jacentes :

$$\text{Rooms\_per\_occupant} = \frac{\text{AveRooms}}{\text{AveOccup}}$$

$$\text{Bedrooms\_per\_room} = \frac{\text{AveBedrms}}{\text{AveRooms}}$$

$$\text{MedInc\_times\_GeoAxis} = \text{MedInc} \times \text{GeoAxis}$$

— **Rooms_per_occupant** : indique le nombre d'individus par pièce, qui illustre le degré de densité d'occupation dans l'habitation.
— **Bedrooms_per_room** : Estime le pourcentage de chambres servant de chambres à coucher ; quand cette proportion s'accroît (moins d'espaces multifonctionnels), la valeur des maisons a tendance à augmenter.
— **MedInc_times_GeoAxis** :Cela témoigne de l'effet fusionné d'un revenu médian élevé et d'une position géographique privilégiée : dans les zones cossues où résident des familles fortunées, les valeurs immobilières ont tendance à être plus élevées.

Ces caractéristiques récentes, dérivées d'une étude méthodique des interrelations dans les données, aident à déchiffrer certains éléments du marché immobilier non saisis par les variables brutes, comme en témoigne l'évolution notable des performances des modèles suite à leur incorporation.

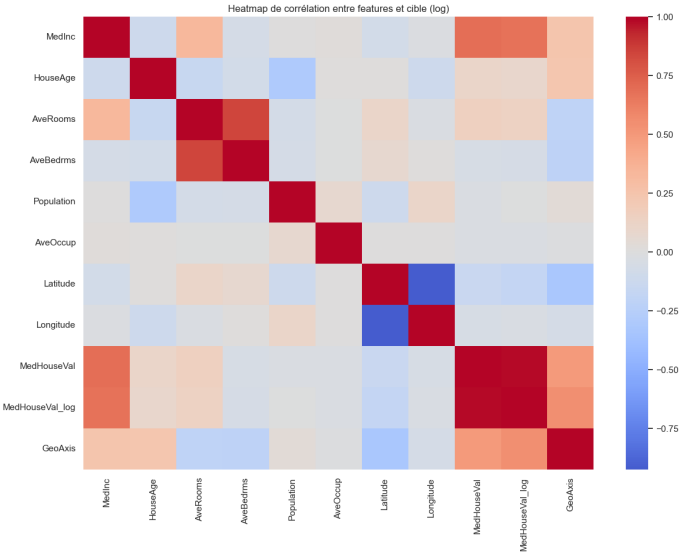

FIGURE 2. Matrice de corrélation et distributions des caractéristiques avant features engineering.

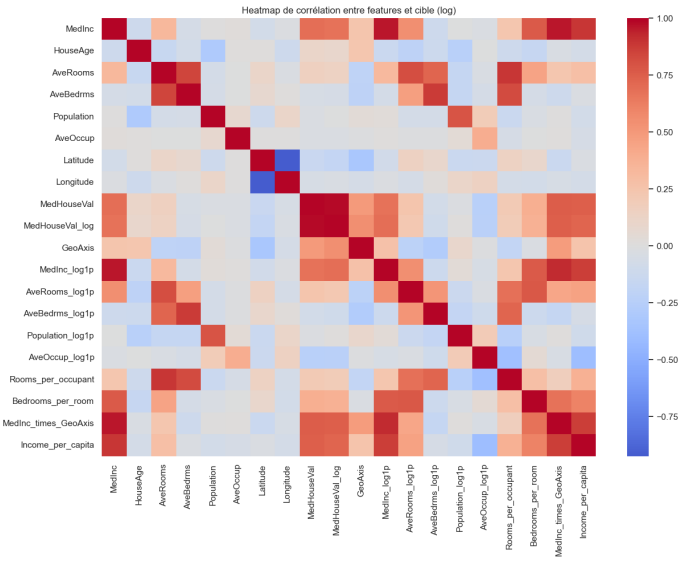

FIGURE 3. Matrice de corrélation et distributions des caractéristiques apres features engineering .

C. Préparation des Données

Les données sont divisées en deux groupes : l'ensemble d'entraînement (80%) et l'ensemble de test (20%). L'*StandardScaler* (normalisation par centering and scaling) est formé uniquement sur le jeu de données d'entraînement afin d'éviter toute fuite d'information. Toutes les caractéristiques sont modifiées pour avoir une moyenne de zéro et un écart-type de un.

D. Modèles Évalués

1) Modèle de Référence (Dummy)

Afin de définir une référence de performance minimale, nous mettons en place un modèle naïf qui anticipe sans cesse la moyenne des prix observés dans le jeu de données d'entraînement :

$$\hat{y}_i = \bar{y} = \frac{1}{N_{\text{train}}} \sum_{j \in \text{train}} y_j \quad \forall i$$

Ce modèle Dummy, configuré en utilisant la stratégie « mean », fait office de benchmark absolu pour juger l'apport des modèles plus élaborés. Toute performance qui ne dépasse pas cette référence indiquerait que le modèle est moins informatif que la simple moyenne historique, alors qu'une performance supérieure montrerait une capacité de prédiction réelle. Cette méthode offre une référence cruciale pour juger de manière objective la contribution des diverses architectures d'apprentissage automatique.

2) Régression Linéaire (Baseline)

La régression linéaire sert de modèle standard pour les problèmes de prédiction continue, établissant une référence de performance grâce à une architecture paramétrique simple. Ce modèle suppose l'existence d'un lien linéaire entre les variables explicatives et la variable dépendante :

$$\hat{y} = \mathbf{w}^\top \mathbf{x} + b$$

où $\mathbf{w} \in \mathbb{R}^d$ Correspond au vecteur de poids lié à chaque caractéristique et $b \in \mathbb{R}$ le terme de biais. L'estimation des paramètres se fait par la minimisation de l'erreur quadratique moyenne, soit en utilisant les équations normales, soit en recourant à la méthode de descente du gradient. Cette méthode offre non seulement des prévisions, mais aussi des aperçus compréhensibles concernant l'importance relative des différentes variables grâce à la taille des coefficients de régression.

3) K-Nearest Neighbors (KNN)

La méthode des K-Nearest Neighbors (KNN) repose sur le principe essentiel de similarité locale et est classée parmi les techniques non-paramétriques. Pour chaque élément à prédire, l'algorithme détermine les k échantillons d'apprentissage les plus proches dans l'espace des caractéristiques, en se basant sur une distance spécifiée. La prédiction découle d'un regroupement des valeurs cibles de ces voisins :

$$\hat{y}_i = \frac{1}{k} \sum_{j \in \mathcal{N}_k(i)} y_j$$

où $\mathcal{N}_k(i)$ fait référence à l'ensemble des k voisins les plus proches. Cette méthode est basée sur l'assumption sous-jacente que les points situés à proximité les uns des autres dans l'espace des caractéristiques possèdent des valeurs cibles semblables. La souplesse du KNN découle de l'absence d'hypothèses paramétriques concernant la forme de la fonction de régression, ce qui permet de saisir des relations complexes sans nécessiter une modélisation explicite.

4) Random Forest (RF)

Le modèle Forêt Aléatoire est une technique d'ensemble qui agrège les prédictions de plusieurs arbres de décision autonomes. Cette structure, au lieu d'un seul arbre, diminue considérablement le danger du surapprentissage tout en saisissant des liens complexes non linéaires grâce à la notion de sagesse collective (wisdom of crowds). Chaque arbre f_t est formé sur un échantillon bootstrap des données d'apprentissage, en choisissant aléatoirement des caractéristiques à chaque scission de nœud afin de garantir la diversité du groupe. La prévision finale compile les résultats de tous les arbres en calculant une moyenne simple :

$$\hat{y} = \frac{1}{T} \sum_{t=1}^{T} f_t(\mathbf{x})$$

L'ajustement des hyperparamètres est une phase cruciale pour optimiser les performances du modèle, étant donné que chacun d'eux régule un aspect spécifique de son fonctionnement. Le paramètre **n_estimators** correspond au nombre d'arbres dans la forêt : un chiffre plus élevé améliore la stabilité du modèle tout en augmentant le coût en calcul. L'option **max_depth** définit la profondeur maximale des arbres ; quand elle est configurée sur *None*, les arbres se développent sans contrainte jusqu'à ce que toutes les feuilles soient pures ou qu'elles respectent le critère **min_samples_split**.

Le paramètre **min_samples_split** détermine le nombre minimum d'échantillons nécessaires pour procéder à la division d'un nœud interne, ce qui a un impact direct sur le niveau de détail de l'arbre. En ce qui concerne **max_features**, ce paramètre détermine le nombre de caractéristiques prises en compte lors de la recherche du meilleur partage ; l'option *log2* choisit $\log_2(d)$ caractéristiques, favorisant ainsi une diminution de la corrélation entre les arbres. Finalement, l'option **bootstrap** détermine si une technique d'échantillonnage bootstrap est mise en œuvre. Quand il est défini comme *False*, chaque arbre est formé sur la totalité des données.

5) Perceptron Multicouches (MLP)

L'architecture du Perceptron Multicouches, un type de réseau de neurones artificiels, permet de reproduire des relations complexes grâce à une succession de transformations non-linéaires. À l'inverse des modèles linéaires,

le MLP est capable d'apprendre des représentations hiérarchiques des données par le biais de ses couches cachées :

$$\mathbf{h}_1 = \text{ReLU}(\mathbf{W}_1\mathbf{x} + \mathbf{b}_1)$$
$$\mathbf{h}_2 = \text{ReLU}(\mathbf{W}_2\mathbf{h}_1 + \mathbf{b}_2)$$
$$\hat{y} = \mathbf{w}_o^\top \mathbf{h}_2 + b_o$$

où $\mathbf{W}_1$, $\mathbf{W}_2$ et $\mathbf{w}_o$ sont les matrices de poids, $\mathbf{b}_1$, $\mathbf{b}_2$ et b_o les biais, et ReLU (Rectified Linear Unit) La fonction d'activation introduit la non-linéarité indispensable. Cette structure est capable de saisir les interactions complexes entre les caractéristiques grâce à la propagation en avant, tandis que la rétropropagation modifie les paramètres afin de réduire l'erreur de prédiction.

E. Optimisation des Hyperparamètres

Pour chaque modèle, nous effectuons une recherche systématique des hyperparamètres optimaux :

— **KNN** :Méthode non-paramétrique basée sur le principe de ressemblance locale. Pour chaque cas à prédire, l'algorithme sélectionne les k exemples d'entraînement les plus proches en se basant sur une mesure de distance préétablie. La prévision découle alors d'une consolidation des valeurs cibles de ces voisins. Après une recherche approfondie, la meilleure configuration choisie fait appel à $k = 11$ voisins, en privilégiant une pondération basée sur l'inverse de la distance ('weights='distance'') et en utilisant la distance de Manhattan ('metric='manhattan', p=1'). Cette approche procure une robustesse face aux variations d'échelle des caractéristiques tout en préservant une précision optimale au niveau local.

— **Random Forest** : Le modèle Random Forest a été perfectionné en procédant à une recherche exhaustive des hyperparamètres, suivie d'une validation croisée stricte. Le paramétrage final après optimisation inclut 317 arbres de décision ('n-estimators=317') avec une profondeur sans limite ('max-depth=None'), ce qui autorise chaque arbre à s'ajuster complètement à la complexité des données. L'option 'min-samples-split=3' assure qu'un nœud ne se scinde que s'il comporte au minimum trois échantillons, évitant de ce fait une suradaptation à des motifs trop particuliers. L'approche 'max-features='log2'' choisit de manière aléatoire environ quatre attributs ($\log_2(18) \approx 4$) à chaque scission, ce qui favorise la variété des arbres. L'option 'bootstrap=False' signifie que chaque arbre est formé sur l'intégralité des données au lieu de se baser sur des sous-ensembles bootstrap, ce qui optimise l'information accessible pour chaque prise de décision.

— **MLP** : Structure de réseau de neurones artificiels constituée de couches denses reliées entre elles. Le modèle acquiert des représentations hiérarchisées des données à travers une série de transformations non linéaires. Après une recherche aléatoire, la meilleure configuration obtenue inclut deux couches cachées de 128 et 64 neurones avec activation ReLU, un taux d'apprentissage constant, une régularisation L2 de $\alpha = 0.001$, et l'optimiseur Adam pour un maximum de 500 itérations. Ce modèle permet de représenter des relations complexes et non linéaires entre les variables d'entrée et la cible, tout en maîtrisant l'apprentissage excessif.

La sélection finale est basée sur la performance sur l'ensemble de validation, mesurée par le MAE.

IV. Résultats Expérimentaux

A. Performances Comparatives

Le tableau I montre les performances relatives des divers modèles sur le jeu de test. Le Random Forest se démarque comme le modèle idéal, affichant un coefficient de détermination $R^2 = 0.821$ et une erreur absolue moyenne de 0.308 (sur l'échelle réelle). Cette performance exceptionnelle est due à sa faculté de saisir des relations complexes non-linéaires tout en restreignant le surapprentissage au moyen de l'agrégation de plusieurs arbres décisionnels.

TABLE I
COMPARAISON DES PERFORMANCES SUR L'ENSEMBLE DE TEST

Modèle	**RMSE**	**MAE**	$\mathbf{R^2}$
Dummy (moyenne)	1.144856	0.906069	-0.000219
Régression Linéaire	0.667910	0.466952	0.659569
KNN	5.330366e-01	3.479302e-01	0.783176
Random Forest	**0.4844**	**0.3084**	**0.8210**
MLP	0.527195	0.347354	0.787902

B. Validation Croisée du Random Forest

Pour évaluer la stabilité du MLP, nous effectuons une validation croisée 5-fold. Les résultats (tableau II) montrent une faible variance entre les folds , confirmant la robustesse du modèle.

TABLE II
RÉSULTATS DE VALIDATION CROISÉE 5-FOLD DU RANDOM FOREST

Fold	**RMSE**	**MAE**	$\mathbf{R^2}$
1	0.506663	0.327159	0.813979
2	0.484107	0.309966	0.823568
3	0.487619	0.309985	0.821705
4	0.510256	0.316130	0.806732
5	0.483978	0.314636	0.818969
Moyenne	0.4945	0.3156	0.8170
Écart-type	0.0129	0.0070	0.0068

V. Discussion

A. Sélection du Modèle Optimal

Parmi les modèles analysés, le Random Forest se distingue comme la solution idéale pour notre enjeu de prévision immobilière. Plusieurs caractéristiques distinctives de cette approche justifient ce choix :

a) **Interprétabilité**L'importance des caractéristiques peut être directement déterminée grâce aux évaluations d'impureté des arbres, fournissant des perspectives exploitables.
b) **Efficacité de calcul** : La formation est nettement plus rapide sans sacrifier la précision.
c) **Robustesse** : Moins sensible à l'initialisation et aux hyperparamètres que le MLP.
d) **Déploiement** : Modèle plus léger et avec moins de dépendances logicielles.

B. Importance des Caractéristiques

L'évaluation de l'importance des caractéristiqu es dans le Random Forest indique que le revenu médian (MedInc) est le facteur prédictif principal, suivi de la caractéristique géographique synthétique (GeoAxis) et du nombre moyen de pièces (AveRooms). Cette structure reflète l'intuition économique : le pouvoir d'achat sur place et l'emplacement constituent les facteurs majeurs influençant les prix de l'immobilier.

C. Limitations et Biais Potentiels

1) Hétéroscédasticité des Résidus

L'accroissement constant des erreurs pour les propriétés de luxe indique que le modèle saisit moins efficacement les dynamiques du marché dans cette catégorie. Des méthodes différentes comme la régression quantile pourraient réduire ce biais.

2) Données Limitées dans le Temps

Les données du dataset remontent à 1990 et ne tiennent pas compte des changements récents sur le marché immobilier en Californie. L'ajout de données historiques permettrait de modéliser les tendances sur le long terme.

3) Géocodage Approximatif

Dans le cadre de l'application de déploiement, l'utilisation du géocodage par API peut entraîner des inexactitudes pour les emplacements ambigus. Un couplage avec des bases de données géographiques plus exactes renforcerait la fiabilité.

D. Comparaison avec la Littérature

Nos performances ($R^2 = 0.8210$) sont comparables ou supérieures à celles mentionnées dans des recherches similaires qui ont employé le même ensemble de données. Pour illustrer, Géron [11] mentionne un R^2 approximativement égal à 0.82 avec une forêt aléatoire optimisée, alors que les méthodes de boosting par gradient obtiennent généralement entre 0.85 et 0.87. L'avantage relatif de notre mise en œuvre peut être attribué à l'ingénierie des caractéristiques avancées, notamment la conception de la caractéristique GeoAxis.

VI. Application de Déploiement

Nous avons conçu une application web interactive avec Gradio pour permettre aux utilisateurs finaux de comprendre facilement la prédiction des prix immobiliers. Cette plateforme offre aux utilisateurs la possibilité d'entrer les spécificités d'une propriété—nombre de pièces, chambres à coucher, ancienneté de l'habitation, situation géographique (ville et quartier), ainsi que le nombre d'habitants—et d'obtenir une estimation du prix médian instantanément. L'application comprend un processus exhaustif de traitement, allant de l'entrée des données à la prévision finale, assurant simultanément une facilité d'utilisation et une solidité technique.

L'application est conçue selon un processus structuré en différentes phases. Premièrement, l'emplacement renseigné (ville et quartier) est transformé en coordonnées géographiques (latitude, longitude) grâce à l'API Geocodio. Ces coordonnées, en association avec les autres attributs saisis, sont par la suite enrichies à travers le même processus d'ingénierie des caractéristiques utilisé durant l'entraînement, englobant le calcul de la variable composite GeoAxis et des taux démographiques. Les 18 caractéristiques obtenues sont normalisées à l'aide du StandardScaler pré-entraîné, puis elles sont présentées au modèle Random Forest sérialisé. Finalement, l'estimation en échelle logarithmique est convertie en valeur réelle (en dollars) et présentée à l'utilisateur. Déployée en tant que serveur web Python léger, l'application garantit une latence de moins de deux secondes. Elle dispose également de systèmes de gestion d'erreurs pour assurer un usage sans accroc, même en présence d'entrées incorrectes ou lors d'une indisponibilité temporaire des services externes.

VII. Conclusion et Perspectives

Cette recherche a prouvé l'efficacité d'une démarche méthodique dans la prévision des prix de l'immobilier en Californie, où le développement systématique des caractéristiques—y compris l'élaboration de la variable composite GeoAxis et de ratios démographiques significatifs—a conduit à une amélioration considérable des performances prédictives. Le modèle Random Forest optimisé, obtenant un coefficient de détermination$R^2 = 0.8210$,s'est avéré

être l'option la plus équilibrée, combinant précision, interprétabilité et efficience computationnelle. L'examen des résidus a révélé des motifs d'hétéroscédasticité persistants, alors que la mise en œuvre d'une application web interactive a confirmé la pertinence pratique de la méthode. Parmi les évolutions futures, on envisage l'intégration de données temporelles, l'étude d'architectures d'ensemble sophistiquées, et l'expansion géographique afin d'examiner la capacité de généralisation des modèles. Ceci permettrait de corroborer l'importance des approches de machine learning appliquées à la résolution de problématiques économiques complexes.

RÉFÉRENCES

[1] F. Pedregosa et al., "Scikit-learn : Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[2] Scikit-learn Developers, "California Housing Dataset Documentation," 2023. [En ligne]. Disponible : https://scikit-learn.org/stable/datasets/real_world.html#california-housing-dataset

[3] R. K. Pace and R. Barry, "Sparse spatial autoregressions," *Statistics & Probability Letters*, vol. 33, no. 3, pp. 291–297, 1997.

[4] B. H. Baltagi and D. Li, "Predicting house prices using spatial data," *Empirical Economics*, vol. 35, no. 2, pp. 309–328, 2008.

[5] Z. Wang, R. C. K. Chan, and M. Zhang, "Machine learning approaches for housing price prediction : A comparative study," *Expert Systems with Applications*, vol. 113, pp. 240–251, 2018.

[6] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[7] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

[8] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. 13, no. 1, pp. 21–27, 1967.

[9] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[10] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society : Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[11] A. Géron, *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*, 2nd ed. O'Reilly Media, 2019. (Section 2 : "End-to-End Machine Learning Project")