De la complexité spatiale aux prix de l'immobilier : approches statistiques

Sarah Soleiman

THÈSE de DOCTORAT

de

l'UNIVERSITÉ PARIS 1 PANTHÉON-SORBONNE

Spécialité : Mathématiques appliquées

présentée par

Sarah SOLEIMAN

pour obtenir le titre de

DOCTEUR DE L'UNIVERSITÉ PARIS 1 PANTHÉON-SORBONNE

Sujet

De la complexité spatiale aux prix de l'immobilier : approches statistiques

Soutenance le 16 janvier 2023 devant le jury composé de :

Rapporteurs :

Annick VIGNES, Directrice de recherche INRAE & Professeure École des Ponts

Thomas ROMARY, Maître de conférences HDR École des Mines

Président du jury :

Denis ALLARD, Directeur de recherche INRAE

Examinateurs :

Liliane BEL, Professeure AgroParisTech

Ndèye NIANG KEITA, Professeure CNAM

Invités :

Julien RANDON-FURLING, CPJ ENS Paris-Saclay – Directeur de thèse

Pierre VIDAL, Directeur scientifique Meilleurs Agents – Encadrant CIFRE

Table des matières

Remerciements

Je suis extrêmement honorée d'avoir pu soutenir ma thèse de mathématiques appliquées en janvier dernier. Je n'aurais pu espérer accomplir ce travail dans de meilleures conditions et je tiens à exprimer ma profonde gratitude à toutes celles et ceux qui par leur accompagnement, leur soutien et leurs conseils, ont rendu cela possible.

Je remercie tout particulièrement mon directeur de thèse, Julien Randon Furling, aux côtés de qui j'ai tant appris et qui, par sa pédagogie, son écoute et sa bienveillance, m'ont permis d'aller au bout de ce projet.

Je remercie également les membres du jury Liliane Bel, Ndèye Niang Keita et les rapporteurs Annick Vignes, Thomas Romary de s'être intéressés à mon travail et de l'avoir enrichi par la pertinence de leurs retours. Je souhaiterais exprimer ma gratitude particulière à Denis Allard pour les échanges fructueux que nous avons pu avoir sur le krigeage. Je suis très honorée qu'il ait accepté de présider mon jury de thèse.

Merci également au laboratoire du SAMM de l'Université Paris 1 Panthéon Sorbonne pour son encadrement et particulièrement à Marie Cottrell pour son aide d'une générosité sans pareil.

Je suis très reconnaissante à Meilleurs Agents de m'avoir accordé l'opportunité de mener ma thèse dans le cadre d'une convention CIFRE et de m'avoir fourni tous les outils pour y parvenir et m'y épanouir.

Je n'aurais pu espérer avoir de meilleurs encadrants que Thomas Lefebvre et Pierre Vidal. Je remercie Thomas de m'avoir poussé à faire une thèse, pour ses relectures, ses conseils, mais surtout sa constante et précieuse bienveillance.

L'un des défis majeurs de toute thèse CIFRE est de trouver le juste équilibre

entre le temps consacré aux prérogatives de l'entreprise et celui que nécessite la recherche académique. Face à cette difficulté, j'ai eu la chance de pouvoir compter sur le soutien sans faille de Pierre qui a toujours veillé à ce que cet équilibre soit respecté. Je le remercie également pour sa rigueur, ses encouragements à élargir mes horizons et aller toujours plus loin dans l'approfondissement de mes recherches.

De manière plus large, je remercie tous les membres de l'équipe Data Science de Meilleurs Agents pour leur bienveillance, les échanges stimulants que l'on a pu avoir et leur aide au quotidien, y compris sous la forme de pâtisseries. Je remercie tout particulièrement Youcef et David qui n'ont jamais hésité à prendre de leur temps pour m'aider et sans qui cette thèse n'aurait pas été réalisable dans le temps imparti.

La thèse CIFRE prend tout son sens lorsqu'au sein de l'entreprise des collaborations étroites se forment et des intérêts pour certains sujets de recherches convergent. À ce titre, je remercie Carmélo avec qui j'ai eu le plaisir de travailler sur la refonte des indices de prix de l'immobilier pendant près de deux ans. Merci aussi à Hadrien d'avoir pris de son temps pour que les réseaux de neurones n'aient plus de secrets pour moi, ce qui a été d'une aide précieuse pour l'élaboration de mon dernier chapitre. Je ne pourrais terminer ces remerciements sans citer Mimile, au risque de lourdes représailles. Merci pour son soutien, son humour et son amitié constante.

Enfin, je remercie mon entourage de m'avoir soutenu et supporté tout au long de ce parcours, en particulier les derniers mois qui ont dû être difficiles.

Mes amis, qui ont toujours cru en moi et qui ont eu la gentillesse de faire semblant de s'intéresser à mes problèmes de modélisations spatiales.

Mes parents, Edna et Henri, qui mériteraient bien plus qu'un paragraphe pour les remercier de tout ce que je leur dois. Leur éducation et leur exemple m'inspirent au quotidien et me donnent la force de croire que tout est possible lorsqu'on s'en donne les moyens. Ce travail n'aurait pu voir le jour et aboutir sans leur amour, leur confiance, leur patience et leurs encouragements, mais aussi le sens de la rigueur et de la résilience qu'ils m'ont transmis. À ce titre, cette thèse est aussi un peu la leur.

Mes frères, Yoni et Gary, sur qui je peux toujours compter.

Enfin, un grand merci à Joseph qui partage ma vie et qui a relu mon manuscrit dans sa totalité, plusieurs fois, alors qu'il pensait avoir laissé les mathématiques derrière lui en quittant le lycée.

Ce titre de Docteur en mathématique est la culmination d'un cycle de huit années d'études. Mais plus qu'un titre, il s'agit de l'accumulation de tant de connaissances, de rencontres, d'échanges et de défis relevés que j'espère pouvoir à présent mettre au service de nouveaux projets.

Introduction

Les liens entre la complexité spatiale d'un tissu résidentiel – en particulier urbain – et les prix des biens sur le marché immobilier afférent posent des défis de modélisation mathématique tout à fait intéressants et originaux. En effet, deux logements présentant exactement les mêmes caractéristiques n'auront généralement pas le même prix en fonction de leur emplacement dans une ville, en raison d'effets de quartier. Ces spécificités locales et même micro-locales se révèlent difficiles à quantifier et/ou à mesurer sans faire entrer en jeu une connaissance intime de la ville en question. Partant, un des enjeux de cette thèse, réalisée au sein de l'entreprise d'estimation immobilière Meilleurs Agents dans le cadre d'une convention industrielle de formation par la recherche (CIFRE), a été d'élaborer des approches statistiques pour cette problématique, formalisée comme suit : étant donnée la carte géographique d'une ville et les transactions immobilières enregistrées dans cette ville jusqu'à un instant t, comment estimer au mieux le prix d'une transaction qui surviendrait à l'instant $t + 1$ en tel ou tel point de la ville ?

On pourrait penser qu'il suffit d'observer les ventes passées d'un même bien pour en tirer une bonne estimation (méthode des ventes répétées), mais malheureusement l'information immobilière relève d'un contexte d'information rare. Il importe de souligner que cela n'est pas dû à un défaut d'accès à telle ou telle base, mais qu'il s'agit là d'une caractéristique structurelle. Pour illustration : il y a approximativement 60 000 immeubles résidentiels à Paris, et on observe chaque année environ 30 000 transactions d'appartements, soit en moyenne une demi-transaction par immeuble et par an — et ceci constitue « le meilleur des cas » en France. Les données de transactions ne suffisent donc pas à estimer le parc directement, surtout pour les zones moins dynamiques. Pour combler le manque, il faut s'appuyer sur des modèles de diffusion d'une information de prix, dans l'espace des biens et dans l'espace géographique.

Cette problématique recouvre ainsi des questions de statistiques économétriques et de processus spatiaux. Premièrement, comment adapter un prix en fonction des caractéristiques du logement : par exemple, comment estimer un appartement comprenant cinq pièces et deux salles de bain à partir d'une transaction antérieure

voisine portant sur un appartement disposant, lui, de deux pièces et une salle de bain ? Deuxièmement, comment diffuser les prix spatialement : l'information de prix apportée par une transaction peut-elle raisonnablement être étendue aux biens alentour – dans quelle ampleur, jusqu'à quelle distance ?

Pour prendre en charge les deux composantes de diffusion (la diffusion dans l'espace des caractéristiques des logements et celle dans l'espace géographique), des approches existent. Elles sont fondées essentiellement sur des méthodes de régressions dites "hédoniques" [43], qui intègrent les effets spatiaux soit sous forme de variables indicatrices de quartiers géographiques prédéfinis, soit en laissant les coefficients de la régression varier spatialement dans un cadre appelé GWR (pour *geographically weighted regression*, régression pondérée spatialement) [9]. En se focalisant plus spécialement sur la dimension spatiale, il existe également des approches fondées sur la méthode géostatistique du krigeage [35].

C'est sur la diffusion spatiale que se concentre cette thèse, en cherchant à améliorer les méthodes existantes, sous une contrainte de faisabilité opérationnelle : l'idée étant, à terme, d'une implémentation dans le cadre industriel.

Un de nos points de départ est le constat que la GWR, telle que couramment utilisée, souffre d'une limite dans son appréhension du tissu urbain : avec les noyaux standard, gaussiens ou autres, elle est isotrope et ne dépend que de la distance géographique. Or, quand on se place en un point d'une ville, à moins de se trouver par un hasard très peu probable à l'épicentre d'un quartier parfaitement isotrope, la typologie du voisinage sera différente selon la direction dans laquelle on s'oriente. Et ce phénomène est évidemment exacerbé sur des frontières de quartiers. En outre, des quartiers, voire des microquartiers, peuvent être géographiquement très éloignés, mais très similaires en termes de tissu urbain. Pour des appartements situés dans des quartiers similaires, mais espacés géographiquement, les prix auront plus de raisons de se ressembler que si l'on considère deux appartements géographiquement proches, mais dont les micro-quartiers ne présentent pas les mêmes qualités de tissu urbain.

Une première piste consiste donc à proposer une mesure de similarité entre les quartiers d'une ville et à l'intégrer dans les méthodes d'estimation par régression ou géostatistiques. Des travaux récents ont justement pu montrer l'intérêt de renouveler les techniques d'analyse statistique pour étudier les (dis)similarités socio-spatiales : citons notamment (Cottrell et al., 2018) [41], qui met en oeuvre des techniques

d'apprentissage et de *clustering*, en l'occurrence l'algorithme SOM (Self-Organizing Maps)[33, 4], pour la détection de structure spatiale dans un cadre multidimensionnel pouvant prendre en compte un grand nombre de variables.

Cette première piste, qui sera suivie dans les chapitres 4 et 5, apporte des éléments de réponse sur la question de la diffusion spatiale de l'information sans changer la structure fondamentale des modèles (GWR ou krigeage par exemple). Elle vise à améliorer les modèles existants en passant par un raffinement suivant lequel on considère qu'un point proche, mais se situant dans un quartier ou micro quartier très différent, pèsera moins qu'un point géographiquement éloigné, mais dans un quartier très semblable. Le quatrième chapitre examine ainsi le croisement de la GWR avec l'algorithme SOM. Notre approche s'effectue en deux temps : nous appliquons d'abord l'algorithme SOM sur un large ensemble de données socio-économiques afin de faire ressortir la structure socio-spatiale. Nous combinons ensuite la distance géographique et la distance sur la carte auto-organisée dans une GWR. Le cinquième chapitre explore quant à lui le croisement de l'algorithme SOM avec le krigeage. En utilisant les outils liés à la géostatistique tels que l'étude du semivariogramme sur les observations de clustering, on apporte de l'information en plus de celle donnée par la distance dans l'espace SOM uniquement. De la même manière que pour le chapitre 4, on applique d'abord l'algorithme SOM afin de regrouper des entités géographiques. Nous multiplions ensuite les covariances SOM et spatiales afin de calculer les poids de krigeage.

Les performances des deux nouvelles approches sont comparées à celles de leurs équivalents de base, c'est-à-dire à la GWR et au krigeage.

Cependant, quand bien même ces nouvelles techniques apporteraient des améliorations notables, elles présentent des limites. En effet, elles nécessitent d'acquérir une autre source de données et de traiter celles-ci pour pouvoir alimenter les modèles déjà existants. Il faut aussi trouver le meilleur moyen d'intégrer cette nouvelle information dans la régression ou le krigeage, ce qui n'a rien d'évident. Nous proposons donc, dans le sixième chapitre, un changement de paradigme. L'idée en est que les prix et la dynamique passée des prix renseignent sur les ruptures géographiques et la similitude de quartiers spatialement éloignés sans avoir besoin de données supplémentaires. Les prix contiennent en fait l'essentiel de l'information socio-spatiale, intégrée par les agents (acquéreurs et vendeurs) qui ont réalisé les transactions. Il suffit donc, en quelque sorte, « d'apprendre » cette information à travers les prix

passés. Considérant alors qu'une ville est un réseau de localisations (de parcelles ou d'immeubles), nous proposons de modéliser ce réseau comme un réseau de neurones, où chaque neurone est caractérisé par le prix au mètre carré de la localisation. Une transaction active un neurone et modifie son prix et celui de ses voisins par répercussion. Les données des transactions passées nous permettent ainsi d'entrainer ce réseau afin qu'il apprenne les dynamiques du marché du logement dans la ville.

Nous montrons que ce paradigme, même dans une première implémentation « naïve » (en utilisant simplement le réseau géographique comme structure de voisinage pour le réseau de neurones), donne, sur les exemples de Paris et Les Lilas, des résultats au moins aussi bons que les méthodes largement utilisées dans la littérature pour l'estimation des prix de l'immobilier (GWR, krigeage). Il sera ensuite possible d'effectuer de l'apprentissage de structure afin de ne pas se restreindre au réseau géographique uniquement (et de se retrouver face aux mêmes limites qu'énoncées plus haut). Ceci fera l'objet d'un travail postérieur à la rédaction de ce manuscrit.

Avant d'entrer dans les trois chapitres introduits ci-dessus, nous prenons le temps dans les chapitres 1, 2 et 3 de présenter les données et les méthodes existantes utilisées dans cette thèse (à savoir les modèles hédoniques, le modèle de GWR et le krigeage) et de passer en revue la littérature s'intéressant au même problème que nous. En outre, dans un septième et dernier chapitre, nous présentons la mise en œuvre d'une nouvelle méthode de création d'indices de prix de l'immobilier couvrant l'entièreté du territoire français, méthode élaborée et mise en production dans le cadre industriel pendant cette thèse : elle utilise la même idée que celle du chapitre 4, à savoir enrichir l'information de prix géographique par une information de similarité obtenue à partir de l'algorithme SOM. Signalons enfin que l'ensemble des trois modèles développés au cours de cette thèse CIFRE ont également pu être testés et appliqués sur des données provenant des bases de Meilleurs Agents dans le cadre de l'élaboration de cartes des prix de l'immobilier.

Chapitre 1

Méthodes existantes utilisées dans cette thèse

Plusieurs méthodes existent pour estimer les prix de l'immobilier, des méthodes dites hédoniques aux réseaux de neurones. Parmi elles, trois ressortent et elles sont décrites dans ce chapitre, ainsi que deux algorithmes classiques de *clustering* (classification) que nous utiliserons ultérieurement pour améliorer les méthodes précitées.

1.1 Régression

De manière assez naturelle, la méthode la plus répandue pour l'étude des prix immobiliers est la régression [47, 27].

Rappelons très brièvement le principe de la régression. Sous sa forme la plus simple, il s'agit de modéliser statistiquement un vecteur de N prix observés (ou plus couramment celui des logarithmes des prix observés) comme une combinaison linéaire de p variables explicatives et d'une constante (appelée intercept). Matriciellement, cela donne l'équation suivante :

$$P = X\beta + \mathcal{E} \tag{1.1}$$

où P est le vecteur-colonne des N (log-)prix observés, X est la matrice de taille $N \times (p+1)$ composée d'une colonne de 1 (pour l'intercept) et des valeurs des p variables explicatives pour chacun des N prix observés, β est le vecteur-colonne des $p+1$ coefficients de la régression (l'intercept plus un coefficient pour chaque

variable explicative), et enfin $\mathcal{E}$ est le vecteur-colonne des N résidus (les différences entre le prix « prédit » par la combinaison linéaire et le prix observé, pour chacune des N observations).

La détermination du vecteur β peut se faire de différentes manières ; la plus standard, appelée méthode des moindres carrés (OLS), consiste à trouver le vecteur $\hat{\beta}$ qui minimise la somme des carrés des résidus :

$$\hat{\beta} = \text{Argmin}_{\beta \in \mathbb{R}^{p+1}} ||P - X\beta||^2 \tag{1.2}$$

Lorsque X est de rang plein (i.e. ses colonnes sont linéairement indépendantes), on peut montrer que :

$$\hat{\beta} = \left(X^T X\right)^{-1} X^T P, \tag{1.3}$$

où X^T est la transposée de la matrice X.

1.1.1 Régression hédonique

Dans le cadre économétrique de la théorie des prix hédoniques, introduite par Rosen en 1974 [43], le choix des variables explicatives permet de mesurer, à travers les coefficients β, la contribution de chaque caractéristique d'un logement à son prix sur le marché immobilier. L'idée fondamentale est que ces caractéristiques génèrent de l'utilité pour l'acheteur et possèdent, chacune, une sorte de prix implicite. Rien n'oblige à se limiter à des variables concernant le type de logement : on peut également introduire dans la régression des variables correspondant à une localisation (par exemple, à Paris, une variable binaire pour chaque arrondissement, ou pour chaque quartier administratif ou autre), ou à une période dans le temps, ou bien encore à une facilité d'accès à tel ou tel équipement. Basu et Thibodeau [7] ont ainsi proposé sept catégories de variables explicatives pour les biens immobiliers. Décrivons ici les trois principales uniquement : les variables caractéristiques du bien au sens strict comme le type (maison ou appartement), la surface, le nombre de pièces, le nombre de salles de bain, l'étage du bien ou bien l'époque de construction ; les variables de localisation ou de voisinage comme le quartier ou l'arrondissement où se trouve le bien, l'environnement économique et social, la proximité aux écoles et aux transports ; enfin, les variables temporelles telles que la date de signature de la promesse de vente ou celle de la transaction chez le notaire ou bien encore la période

pendant laquelle les informations sur le bien ont été recueillies. En désignant par C l'ensemble des indices des variables explicatives appartenant à la première catégorie, S pour la deuxième et T pour la troisième, on a donc :

$$\log P_i = \beta_0 + \sum_{k \in C} \beta_k X_k + \sum_{k \in S} \beta_k X_k + \sum_{k \in T} \beta_k X_k + \varepsilon_i. \tag{1.4}$$

Soulignons enfin que, comme dans toute régression, la définition des valeurs nulles des variables constitue le choix d'une situation de référence, en l'occurrence donc d'un bien standard dont le prix sera donné par l'intercept ($\hat{\beta}_0$), tandis que les autres coefficients donneront la différence de prix associée à telle ou telle déviation par rapport au bien standard. Par exemple, si on définit comme bien standard un appartement de deux pièces, alors le coefficient lié à la variable binaire « une pièce » s'interprète comme l'effet marginal (c'est à dire, l'écart à la valeur du bien de référence) du fait d'avoir une pièce plutôt que deux.

Du point de vue des effets spatiaux, qui nous intéressent ici tout particulièrement, la régression hédonique présente une limite importante : elle prend l'espace en compte à travers l'ajout de variables binaires ou catégorielles correspondant à une carte prédéfinie de quartiers. De plus, les coefficients de régression sont universels : quel que soit l'endroit où se trouve une maison, par exemple, la hausse de prix marginale associée à une chambre supplémentaire est fixe. Cependant, il existe des effets locaux qui impliquent une demande différente, et donc une valorisation différente selon les caractéristiques : la valeur ajoutée d'une chambre supplémentaire pourrait être plus grande dans un quartier peuplé de familles avec enfants où l'espace supplémentaire est susceptible d'être considéré comme très avantageux que dans un quartier peuplé de célibataires, pour qui un espace supplémentaire pourrait être considéré comme un élément peu attractif. Dans ce contexte, l'hypothèse d'uniformité spatiale de l'effet des variables explicatives sur la variable dépendante n'est donc pas vérifiée et appliquer une régression hédonique globale sur tout l'espace masquera les variations locales [9]. Une manière de contourner cet écueil consiste à réaliser non plus une régression globale en incorporant des variables représentant l'espace, mais à réaliser autant de régressions que le nombre de points où l'on veut prédire un prix, en tenant compte à chaque fois de tous les prix observés, mais en les

pondérant par la distance au point choisi. Cette méthode de modélisation géolocalisée s'appelle la régression pondérée spatialement et a été introduite par Brundson, Fotheringham et Charlton en 1996 [9] ; nous la présentons dans le paragraphe suivant.

1.1.2 Régression pondérée spatialement : GWR

La modélisation géolocalisée des prix de l'immobilier se justifie par l'observation que les prix sont très corrélés au sein de zones géographiques, de la rue jusqu'au département, et même parfois plus loin encore [44].

Les régressions pondérées spatialement permettent ainsi de prendre en compte l'hétérogénéité spatiale de l'effet des caractéristiques sur la variable d'intérêt (ici, le prix) : par exemple, la surcote des petites surfaces sera plus importante à proximité d'une université qu'ailleurs dans une ville, ou l'impact d'un balcon plus important en bord de mer que si l'appartement donne sur une autoroute.

La GWR (*geographically weighted regression*) consiste à réaliser une régression classique en chaque point, en pondérant les observations en fonction de leur distance au point considéré : les coefficients β décrits plus haut pour la régression ne seront donc plus uniques, mais propres à chaque point de l'espace. On écrit alors la régression au point $\mathbf{u}$ comme :

$$P(\mathbf{u}) = X\beta(\mathbf{u}) + \mathcal{E}(\mathbf{u}), \tag{1.5}$$

autorisant ainsi les coefficients à varier continûment avec la position spatiale.

Bien sûr, la régression porte toujours sur les prix observés en N points $(\mathbf{u}_1, \ldots, \mathbf{u_N})$ et les valeurs observées des p variables explicatives en ces points. Simplement, on cherche cette fois à minimiser

$$\sum_{i=1}^{N} W(\mathbf{u}_i - \mathbf{u})\,\varepsilon_i^2, \tag{1.6}$$

où W est un « noyau », par exemple gaussien :

$$W\left(\mathbf{v}-\mathbf{u}\right)=\exp\left(-\frac{\left|\left|\mathbf{v}-\mathbf{u}\right|\right|^{2}}{2\,\sigma^{2}}\right), \tag{1.7}$$

avec σ un paramètre jouant le rôle d'une distance caractéristique au-delà de laquelle l'influence d'une observation devient quasi négligeable pour le prix au point étudié. De manière générale, on peut montrer [12] que l'estimation du vecteur $\beta(u)$ est donnée par une simple adaptation de la formule classique (équation 1.3), en incorporant le noyau :

$$\hat{\beta}(\mathbf{u})=\left(X^{T}W_{\mathbf{u}}X\right)^{-1}X^{T}W_{\mathbf{u}}P, \tag{1.8}$$

avec $W_{\mathbf{u}}$ est la matrice diagonale $N\times N$ définie par diag $\left(W\left(\mathbf{u}_{1}-\mathbf{u}\right),\ldots,W\left(\mathbf{u}_{N}-\mathbf{u}\right)\right)$. La GWR valorise donc un appartement proche géographiquement de la localisation à estimer via les poids donnés par le noyau W. Parmi les autres méthodes géolocalisées, il y a celles qui appartiennent au champ dit de la géostatistique, qui s'intéressent à l'interaction des observations entre elles.

1.2 Outils de la géostatistique

Le champ d'application de la géostatistique est initialement très restreint, quasi limité à la recherche minière et développé par l'ingénieur minier Danie G. Krige. Les travaux de Georges Matheron [25] formalisent les concepts et le cadre théorique. La géostatistique est aujourd'hui utilisée pour analyser des phénomènes spatiotemporels dans des domaines tels que la biologie [42] ou la météorologie [5].

Avant d'introduire plus en détail ces outils, exprimons simplement l'idée centrale. Il s'agit, dans notre cadre, de prédire le prix P^{*} au point $\mathbf{u}$ directement à travers une moyenne pondérée des observations dont on dispose en N points $(\mathbf{u}_{1},\ldots,\mathbf{u_{N}})$:

$$P^{*}(\mathbf{u})=\sum_{i=1}^{N}\lambda_{i}P(\mathbf{u}_{i}), \tag{1.9}$$

où les poids λ_{i} (appelés poids de krigeage) satisfont $\sum_{i=1}^{N}\lambda_{i}=1$.
L'objectif de la méthode de krigeage est de déterminer les poids permettant de

minimiser

$$\mathbb{E}\big[(P^*(\mathbf{u}) - P(\mathbf{u}))^2\big], \tag{1.10}$$

c'est-à-dire de minimiser l'espérance du carré de l'écart entre la prédiction et la valeur réelle. L'analyse de variogrammes et d'autres outils sont mobilisés pour atteindre cet objectif.

1.2.1 Variogrammes

On note $p(u)$ la variable régionalisée et $P(u)$ la fonction aléatoire (ici le prix du bien) où u désigne la localisation. L'inférence est rendue difficile, car contrairement au cas classique, en statistiques spatiales on ne dispose que d'une réalisation unique du phénomène régionalisé. Pour pallier ce problème, on suit l'explication donnée par Georges Matheron [25] : « Pour que l'inférence soit possible, il est nécessaire d'introduire des hypothèses supplémentaires sur la fonction aléatoire $P(u)$ de façon à réduire le nombre des paramètres dont dépend sa loi. Tel est le but de l'hypothèse stationnaire que nous allons définir : une fonction stationnaire se répète en quelque sorte elle-même dans l'espace, et cette répétition rend à nouveau possible l'inférence statistique à partir d'une réalisation unique ».

Parmi les hypothèses sur la stationnarité, on introduit celles sur la stationnarité d'ordre 2 et intrinsèque en suivant le cours d'introduction à la géostatistique donné par Denis Allard [1] et le manuel d'analyse spatiale de l'INSEE [22].

Definition 1.2.1.1 (Stationnarité d'ordre 2)**.** *La fonction aléatoire $P(.)$ est stationnaire d'ordre 2 si sa moyenne et sa covariance existent et sont invariantes par translation :*

$$\mathbb{E}\big[P(u)\big] = \mu \ \forall u \in \mathbb{R}^2$$
$$Cov\big[P(u+h), P(u)\big] = C(h) \ \forall u \in \mathbb{R}^2$$

où h est le vecteur de décalage entre deux points dans $\mathbb{R}^2$ et $C(h)$ la fonction de covariance de $P(.)$

Definition 1.2.1.2 (Variogramme)**.** *Le variogramme d'une fonction aléatoire $P(.)$*

stationnaire d'ordre 2 est donné par :

$$\gamma(h) = \frac{1}{2}\mathbb{E}\big[(P(u+h) - P(u))^2\big]$$

Les propriétés liées au variogramme sont les suivantes :

- $\gamma(0) = 0$
- $\gamma(h) = \gamma(-h)$
- $\gamma(h) \geq 0$
- $\gamma(h) = C(0) - C(h)$
- $\lim_{||h||\to\infty}\gamma(h) = C(0) - \lim_{||h||\to\infty}C(h) = C(0)$

S'il existe une structure spatiale, des paires d'observations proches géographiquement auront une variance inférieure à celle d'observations géographiquement éloignées. La valeur du variogramme pour des paires extrêmement proches est censée être faible, au contraire pour des paires d'observations géographiquement très éloignées il n'y a pas de dépendance spatiale et leur variance tend vers la variance du processus sous-jacent.

L'hypothèse de stationnarité d'ordre 2 étant trop restrictive et le variogramme ne dépendant que du décalage h, Georges Matheron[25] propose la notion de stationnarité intrinsèque qui permet d'avoir un variogramme pour une plus grande classe de fonctions aléatoires.

Definition 1.2.1.3 (Stationnarité intrinsèque)**.** *La fonction aléatoire $P(.)$ est intrinsèquement stationnaire si ses accroissements sont stationnaires d'ordre 2, c'est-à-dire si :*

$$\mathbb{E}[P(u) - P(u+h)] = 0$$
$$\frac{1}{2}\mathbb{E}\big[(P(u+h) - P(u))^2\big] = \gamma(h)$$

$\forall u, h \in \mathbb{R}^2$

L'hypothèse de stationnarité d'ordre 2 étant plus forte, elle entraine la stationnarité intrinsèque, mais la réciproque est fausse. On énonce ici une propriété du variogramme qui permet d'avoir cette réciproque, et qui nous sera utile dans nos travaux.

Proposition 1.2.1. *Si une fonction aléatoire $P(x)$ vérifie les hypothèses de stationnarité intrinsèque et que son variogramme est borné alors $P(x)$ est une fonction aléatoire stationnaire d'ordre 2.*

On ne considère ici que le cas du variogramme isotropique, il n'y a pas d'effets directionnels et h est la distance séparant deux points $u_i, u_j \in \mathbb{R}^2 \ \forall i, j$.

On dispose des échantillons $P(u_1), ..., P(u_N)$, le **variogramme empirique** est défini par :

$$\hat{\gamma}(h) = \frac{1}{2N(h)} \sum_{i,j:h-\Delta h \leq |u_i - u_j| \leq h+\Delta h} \left(P(u_i) - P(u_j)\right)^2 \tag{1.11}$$

où $N(h)$ est le nombre de paires (u_i, u_j) telles que $|u_i - u_j| \in [h - \Delta h, h + \Delta h]$ et Δh est choisi selon la granularité de la localisation.

Le variogramme empirique n'est pas utilisable en tant que tel, car il n'est pas défini pour n'importe quelle distance et il ne respecte pas toutes les contraintes définies plus haut. Il faut alors ajuster un **variogramme théorique** au variogramme empirique.

Il y a deux types de variogrammes : les bornés et les non bornés. Le variogramme borné est croissant jusqu'à un certain palier. On appelle la valeur de h où $\gamma(h)$ se rapproche de son asymptote la portée. Les paires d'observations séparées par une distance inférieure à la portée sont autocorrélées spatialement alors qu'au-delà de cette distance elles ne le sont plus.

Par définition, $\gamma(0) = 0$ seulement en réalité on observe des discontinuités à l'origine pour des valeurs très proches de zéro, on appelle cette valeur la pépite. Par exemple, deux appartements situés dans le même immeuble et présentant des caractéristiques exactement identiques n'ont pas forcément le même prix.

Parmi tous les modèles existants, on montre ici deux modèles de variogrammes : le modèle exponentiel et le modèle puissance. Le modèle exponentiel est défini par :

$$\gamma(h) = \begin{cases} 0 & h = 0 \\ c_0 + (c_s - c_0)[1 - \exp(-\frac{h}{a})] & h > 0 \end{cases} \tag{1.12}$$

où $c_0 \geq 0$ est la pépite, $c_s > 0$ est le palier et $a > 0$ est la portée.
Le modèle puissance est défini par :

$$\gamma(h) = \begin{cases} 0 & h = 0 \\ c_0 + bh^p & h > 0 \end{cases} \tag{1.13}$$

où $c_0 \geq 0$ $b > 0$ est la pente et $p > 0$. Ce variogramme n'est pas borné, il ne présente pas de palier ni portée.

L'estimation paramétrique du modèle de variogramme peut se faire par moindres carrés ordinaires (MCO), pondérés (MCP) ou par maximum de vraisemblance (MV). On définit M comme le maximum de la distance à considérer (on prend généralement $1/2$ du diamètre du domaine d'étude) on l'appelle pour notre étude le **cut off**. On discrétise l'intervalle $[0, M]$ en un nombre de classes K avec un pas de discrétisation constant $\frac{M}{K} = \delta$ (appellé **lag step**). On décrit ici l'approche par MCO :

$$\hat{\theta}_{MCO} = \arg\,\min_{\theta \in \Theta} \sum_{i=1}^{K} \left(\hat{\gamma}(h_i) - \gamma(h_i; \theta)\right)^2 \tag{1.14}$$

où θ est le vecteur de paramètres de la fonction γ choisi et où $h_i = i\delta$, $i = 1, ..., K$.

Le variogramme fournit une information sur la corrélation entre les observations selon la distance qui les sépare géographiquement. Il est utilisé pour faire de l'interpolation et prédire la valeur de $P(u_0)$ en un point non observé. C'est la méthode de krigeage, formalisée par Georges Matheron[25] et dont le nom vient de l'ingénieur Danie G. Krige.

1.2.2 Krigeage

Dans la suite on suppose que la fonction aléatoire $P(.)$ est intrinsèquement stationnaire, de variogramme $\gamma(h)$ et de moyenne m.

On dispose de N observations $P(u_1), P(u_2), ..., P(u_N)$ de localisations $u_1, u_2, ..., u_N$. L'estimation donnée par le krigeage ordinaire de $P(u_0)$ est de la forme :

$$P^*(u_0) = \sum_{i=1}^{N} \lambda_i P(u_i) \tag{1.15}$$

avec

$$\sum_{i=1}^{N} \lambda_i = 1$$

pour qu'il soit sans biais.

Pour estimer les poids de la combinaison linéaire, on minimise $\mathbb{E}\big[\big(P^*(u_0) - P(u_0)\big)^2\big]$ sous la contrainte $\sum_{i=1}^{N} \lambda_i = 1$, en utilisant la méthode des multiplicateurs de Lagrange.

Les poids λ pour la localisation u_0 sont donnés par le calcul matriciel suivant :

$$\begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_N \\ m \end{bmatrix} = \begin{bmatrix} \gamma(u_1 - u_1) & \cdots & \gamma(u_1 - u_N) & 1 \\ \gamma(u_2 - u_1) & \cdots & \gamma(u_2 - u_N) & 1 \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(u_n - u_1) & \cdots & \gamma(u_N - u_N) & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} \gamma(u_0 - u_1) \\ \gamma(u_0 - u_2) \\ \vdots \\ \gamma(u_0 - u_N) \\ 1 \end{bmatrix} \tag{1.16}$$

1.3 Outils de classification

Nous présentons maintenant rapidement des méthodes de classification (*clustering*) que nous appliquerons à des données démographiques et socio-économiques (les données INSEE) afin d'obtenir une typologie de quartiers. Cette typologie sera utilisée par la suite pour chercher à améliorer les méthodes existantes d'estimation présentées précédemment (GWR et krigeage).

1.3.1 Algorithme SOM

Les cartes auto-organisatrices (Self-Organizing Map : SOM) ont été introduites par Teuvo Kohonen en 1984 [32] et sont de ce fait également appelées cartes de Kohonen. L'algorithme SOM possède un large champ d'applications, depuis le domaine biomédical, notamment pour les méthodes de diagnostic, jusqu'au traitement du signal, en passant par la reconnaissance d'images. Il est ici utilisé en tant qu'algorithme de *clustering*.

L'algorithme SOM est un réseau de neurones qui peut être vu comme réalisant une projection non linéaire de l'ensemble des données sur une grille bidimensionnelle : la carte auto-organisatrice, ou carte de Kohonen. L'intérêt d'utiliser cet algorithme plutôt que celui des centres mobiles (*k-means*) est que la projection conserve la topologie de l'espace d'entrée. Cela veut dire que des observations voisines dans l'espace d'entrée sont projetées dans le même cluster ou dans des clusters voisins sur la carte de Kohonen. En fait l'algorithme *k-means* peut être vu comme un cas particulier de l'algorithme SOM où l'on supprime tout lien de voisinage entre les clusters.

On fixe a priori le nombre de clusters, disons K. La carte de Kohonen est alors un réseau bidimensionnel de K unités (ou neurones) organisés en une grille de K_1 lignes et K_2 colonnes (avec $K_1 \times K_2 = K$). On note $\mathbb{K} = \{1, ..., K\}$ l'ensemble des unités et on définit une distance $d(k, l)$ sur la carte entre deux unités k et l, choisie ici comme la longueur en pas du plus court chemin de l'unité k à l'unité l sur la grille. À chaque unité on associe un vecteur $m_k \in \mathbb{R}^q$, dit protoype, de la même dimensionnalité que les vecteurs des données : q, si on dispose par exemple de n observations q-dimensionnelles, $y_j \in \mathbb{R}^q$, $j = 1, ..., n$. Au temps $t = 0$, les prototypes sont initialisés aléatoirement (tirés dans une distribution pertinente) et notés $m(0) = (m_1(0), m_2(0), ..., m_K(0))$.
L'algorithme SOM est ensuite défini de manière itérative comme suit :

- à l'itération $t + 1$, une nouvelle donnée $y(t + 1)$ est choisie aléatoirement
- étape d'affectation : l'unité gagnante est définie par :

$$k_0(t + 1) = \text{argmin}_{k \in \mathbb{K}} ||y(t + 1) - m_k(t)||^2$$

- étape de mise à jour : l'unité gagnante est activée, son vecteur prototype est dirigé vers le vecteur d'entrée présenté au réseau. Les vecteurs prototypes

voisins de l'unité gagnante sont aussi mis à jour. Ils se dirigent donc vers le vecteur d'entrée, mais généralement dans une moindre proportion.

$$m_k(t+1) = m_k(t) + \alpha(t) h_t(k, k_0)(y(t+1) - m_k(t))$$

où $\alpha(t)$ est un paramètre d'apprentissage à valeurs dans $[0, 1]$, décroissant ou constant au fil des étapes, et h_t est une fonction de voisinage décroissante avec la distance (et éventuellement aussi avec t) et telle que $h_t(k, k) = 1$.

Deux fonctions de voisinage sont couramment utilisées dans la littérature. La première est une fonction binaire *step* qui vaut 1 si la distance entre deux unités k et l est inférieure à un certain rayon sur la grille et 0 sinon. La deuxième est un noyau gaussien :

$$h_t(k, l) = \exp\left(-\frac{d(k, l)^2}{2\, r_{\mathrm{SOM}}^2(t)}\right) \tag{1.17}$$

où r_{SOM} peut diminuer au cours du temps. Souvent le rayon est choisi assez large à l'initialisation pour que la quasi-totalité des unités soient mises à jour par les premières données. Puis, à la fin de la période d'entrainement, le rayon est tel que seule l'unité gagnante est mise à jour.

Lorsque l'entrainement est terminé, pour tout $k_0 \in \mathbb{K}$, le cluster C_{k_0} est défini par la méthode des plus proches voisins comme l'ensemble des données de l'espace d'entrée les plus proches de m_{k_0} que de tout autre prototype :

$$y \in C_{k_0} \Leftrightarrow ||y - m_{k_0}|| = \min_{k \in \mathbb{K}} ||y - m_k|| \tag{1.18}$$

L'ensemble des clusters $C_1, ...C_K$ fournit alors une représentation de l'espace d'entrée (celui des données). On peut également définir une distance SOM entre les vecteurs d'entrée y et y' en remplaçant chaque vecteur par le prototype de son cluster :

$$d_{SOM}(y, y') = ||m_{k_0(y)} - m_{k_0(y')}|| \tag{1.19}$$

Comme l'Analyse en Composantes Principales (ACP), l'algorithme SOM est un bon outil de visualisation de données en grandes dimensions. En effet elle permet de représenter les observations directement via le prototype de leur cluster et ainsi d'étudier les ressemblances entre clusters.

Lorsque le nombre de clusters est trop important, l'analyse par l'oeil humain des cartes géographiques coloriées selon les clusters construits devient impossible. Afin de faciliter l'étude de ces cartes, on effectue une Classification Ascendante Hiérarchique des vecteurs prototypes pour regrouper les clusters en Super-Classes. La fonction utilisée pour produire ces Super-Classes dans ce manuscrit est issue du package **SOMbrero** (du logiciel **R**) et s'appelle superClass. Le principe de la CAH est expliqué dans le section suivante.

1.3.2 Classification ascendante hiérarchique (CAH)

Initialement, chaque observation est classée dans son propre cluster. Si n est le nombre total d'observations alors on a $L = n$ clusters. Toutes les distances entre clusters sont calculées (il y en a $\frac{n(n-1)}{2}$). Les deux clusters les plus proches sont fusionnés (une seule fusion à la fois). Une fois cette étape passée, on recalcule une distance entre les nouveaux clusters, définie par :

$$d(A, B) = \max\{d(a, b), a \in A, b \in B\} \tag{1.20}$$

où A, B sont deux clusters.
Et ainsi de suite, l'algorithme s'arrêtant lorsqu'il n'y a plus qu'un seul cluster ($L = 1$).

Le dendrogramme est la représentation graphique de la CAH qui permet de montrer toutes les fusions. On peut alors choisir une partition en tronquant le dendrogramme en fonction du nombre de clusters souhaité. Lorsque les données contiennent une structure claire en termes de classes d'objets similaires, cette structure est souvent restituée par le dendrogramme dans des branches distinctes. Une hauteur de coupe est pertinente si elle se trouve entre 2 noeuds dont les hauteurs sont relativement éloignées. Cette méthode cherche à minimiser l'inertie intra-classe afin d'obtenir des classes les plus homogènes possible.

Lorsque l'algorithme SOM converge, on constate un regroupement de clusters voisins sur la carte SOM par les Super-Classes issus de la CAH. On peut alors effectuer des statistiques descriptives classiques et comprendre la structure de chaque Super-Classe.

Chapitre 2

Revue de littérature

De nombreuses études ont été menées afin de modéliser les prix des logements et trois grandes familles de méthodes se dégagent dans la littérature : la première repose sur des techniques d'économétrie, la deuxième est basée sur la géostatistique, enfin la dernière tire profit des avancées récentes des modèles de machine learning, comme les réseaux de neurones, en les appliquant à l'estimation immobilière.

2.1 Régression

Les régressions hédoniques sont initialement utilisées pour l'estimation des prix de l'immobilier en incorporant des variables de voisinage pour tenir compte de la spatialité [27, 37, 50]. Dubin et Sung (1990) utilisent la régression hédonique pour déterminer quel ensemble de variables de voisinage explique le mieux la variation des prix des logements [20]. Leurs résultats montrent que la catégorie socioprofessionnelle des voisins (le niveau de revenu, le niveau de l'éducation et la profession) apparaît plus importante que la qualité des services publics (la qualité de l'école ou le niveau de sécurité). Si les modèles hédoniques sont largement appliqués dans le domaine de l'immobilier, cette méthode présente néanmoins des limites comme le choix [28] et le nombre élevé [38] des variables de voisinage pour expliquer les valeurs liées à la localisation, et comme expliquer plus haut, elle ne prend pas en compte le phénomène de non-stationnarité [17] et d'hétérogénéité spatiale [3, 29, 49]. La GWR est donc par la suite un des modèles les plus utilisés dans la littérature pour estimer les prix de l'immobilier. Notamment, McCluskey *et al.*, en 2013 évaluent et

analysent un réseau de neurones artificels (ANN), un modèle SAR (Simultaneous autoregressive models), une régression hédonique et une GWR sur un échantillon de 2694 transactions de logements résidentiels [36]. Ils montrent que la GWR est supérieure en termes d'explicabilité, de fiabilité et de précision du modèle. L'approche par ANN surpasse la régression hédonique en termes de pouvoir prédictif, et donc de précision d'évaluation et approche la performance de la GWR. Cette méthode offre donc le meilleur équilibre entre performance et transparence de la méthodologie. Bitter *et al.*, en 2007, étudient le phénomène d'hétérogénéité spatiale en Arizona à l'aide de la méthode d'expansion spatiale et de la GWR [23, 12], l'application de la régression pondérée géographiquement donne de bien meilleurs résultats que la méthode d'expansion spatiale [8]. Plus récemment, Doumpos *et al.*, en 2021, proposent une analyse comparative des méthodes de régression (paramétriques et non paramétriques) pour l'évaluation du prix d'un bien [19]. Des approches telles que la régression hédonique, la GWR et le krigeage sont appliquées sur une large échantillon de propriétés en Grèce sur la période 2012-2016. Les résultats montrent que la GWR fournit les meilleures performances, surpassant les approches de machine learning qui ne tiennent pas compte des effets spatiaux. Une vue d'ensemble et détaillée des modèles économétriques est présentée dans [2], [11] et [34].

2.2 Géostatistique

Parmi l'application des méthodes issues de la géostatistique, citons d'abord «Prediction of Housing Location Price by a Multivariate Spatial Method : Cokriging» [14], dans lequel l'auteur étudie les modèles de krigeage et cokrigeage dans un contexte générique d'évaluation en masse des objets immobiliers, en particulier dans les zones pauvres en information. D'autres travaux comparent les méthodes géostatistiques à des approches alternatives en études urbaines. Par exemple, Basu et Thibodeau [7] étudient les données de transaction de maison à Dallas aux États-Unis et comparent un modèle de krigeage à une régression hédonique. Ils concluent que les prix de l'immobilier sont substantiellement corrélés dans tous les sous-marchés étudiés. Cependant, contrôlée par la taille de la maison et son âge, l'autocorrélation spatiale est fortement réduite dans de nombreux sous-marchés. Ils remarquent aussi que le modèle de krigeage présente de meilleures performances qu'une régression par

les moindres carrés ordinaires (OLS) dans des sous-marchés avec une forte autocorrélation spatiale, alors qu'un modèle de régression OLS présentera de meilleures performances dans un marché présentant une faible autocorrélation spatiale. Finalement, les auteurs recommandent de détecter s'il existe une structure spatiale pour les prix de l'immobilier, dans un marché donné, en estimant des variogrammes et en contrôlant leur stabilité et leurs formes.

Dans un autre article intitulé « Anisotropic spatial autocorrelation in single-family house prices and in hedonic house-price equation residuals », Gillen *et al.* s'intéressent à des variogrammes directionnels [26]. Les auteurs utilisent des données de transaction à Philadelphie (États-Unis). Ils montrent qu'après avoir effectué un modèle de régression, les résidus du modèle restent spatialement autocorrélés. De plus, ils ont constaté une anisotropie géométrique dans plusieurs sous-marchés immobiliers. Une autre recherche sur le sujet des variogrammes directionnels est le travail de Atkinson et Lloyd[6]. En effet, cet article présente des réflexions sur l'estimation locale de variogrammes à partir de données cartographiques (altitude par exemple) et l'impact positif de cette technique sur la qualité de l'interpolation. Une référence dans ce domaine de recherche est le travail présenté par Dubin en 1998 [21]. L'auteur y discute plusieurs modèles de krigeage et les teste sur des données de transactions à Baltimore (États-Unis). Il propose un modèle de maximum de vraisemblance qui permet de faire la régression hédonique et le krigeage dans le cadre du même modèle, ce qui permet d'améliorer la qualité de l'estimation. De plus il compare plusieurs mesures de distance dans le modèle de krigeage : une distance euclidienne *vs* une mesure du nombre d'immeubles séparant les points. Ils concluent sur la meilleure qualité du modèle utilisant la distance euclidienne.

Une des rares recherches sur le marché immobilier français a été présentée par Simon et Srikhum [45]. Les auteurs étudient plusieurs approches géostatistiques appliquées au marché immobilier parisien. Ils mettent en cause l'hypothèse de stationnarité spatiale des prix immobiliers et proposent plusieurs stratégies de stationnarisation. Ils développent un modèle hybride mêlant une approche géostatistique et une approche hédonique. La plupart des recherches sur les modèles géostatistiques appliquées au prix de l'immobilier postulent en effet que les prix sont stationnaires ou proposent une modélisation basique de la tendance des prix. Dans cette recherche, les auteurs observent que la structure spatiale des zones urbaines modernes est complexe, que l'hypothèse de stationnarité ne peut-être vérifiée et qu'il convient donc de développer un modèle adaptatif en fonction de la structure de la zone estimée.

D'autres études proposent des modèles hybrides, notamment dans «Does my house have a premium or discount in relation to my neighbors ? A regression-kriging approach», les auteurs utilisent la régression hédonique pour estimer la contribution des caractéristiques propres du bien, et le krigeage pour capturer la partie spatiale en calculant la combinaison linéaire des résidus de la régression [15]. Une prime ou une remise est obtenue, qui est un poids supérieur à un (prime) lorsque les caractéristiques structurelles de la propriété en question sont supérieures à celles des propriétés résidentielles voisines et entre zéro et un (remise) lorsqu'elles sont inférieures.

2.3 Machine Learning

Les méthodes d'économétrie ou de géostatistique, et celles issues du machine learning, comme les réseaux de neurones, sont différentes selon leurs cibles : les régressions hédoniques, par exemple, sont des modèles explicatifs, interprétables et moins volatiles qui permettent de répondre à de nombreux enjeux économiques, sociaux et environnementaux, alors que les modèles d'apprentissage automatique sont très souvent moins interprétable, qualifiés de « boîtes noires » [51] et plus volatiles mais ils offrent souvent une capacité prédictive plus puissante que les régressions hédoniques [18, 48, 40]. Bien que les modèles de machine learning aient une forte capacité de prédiction, il est néanmoins essentiel d'y incorporer des effets spatiaux pour pouvoir estimer les prix de l'immobilier de manière optimal comme le montrent Tchuente et Nyawa dans «Real estate price estimation in French cities using geocoding and machine learning» [46] .

D'autres traveaux intègrent directement les effets spatiaux via les coordonnées géographiques. Par exemple Chen *et al.* [13] proposent une nouvelle structure de réseau de neurones pour la prédiction spatiale en ajoutant au vecteur d'entrée une couche d'intégration de coordonnées spatiales via des fonctions de base. Ils montrent que leur méthode, appelée *DeepKriging*, présente de multiples avantages par rapport au krigeage (pour le cas des processus non stationnaires notamment) et au réseaux de neurones profonds (DNN) classiques.

Une autre recherche en machine learning appliquée à l'immobilier compare quatre modèles : la méthode d'expansion spatiale (SEM), une régression à fenêtre glissante (MWR), une GWR et une méthode de filtrage des coefficients de régression via les

vecteurs propres (Eigenvector Spatial Filtering : ESF) [30]. Les auteurs montrent que la méthode ESF présente l'avantage de ne pas lisser les particularités locales (par opposition à la GWR) et est moins pénalisée par la multicolinéarité des variables, seulement cette méthode à tendance à sur apprendre, en plus d'être moins intuitive que la GWR ou la MWR.

Dans un autre article, Del Giudice *et al.* [16], les auteurs développent un réseau de neurones artificiel avec une approche Bayésienne et le testent sur un petit échantillon de transactions immobilières (65 données). La distribution de sortie est calculée en faisant une intégration (sur l'espace des poids) à l'aide de la méthode de Monte Carlo (Markov Chain Hybrid Monte Carlo). Ils montrent que leur nouvelle méthode, MCHMCM, donne l'erreur de prédiction la plus basse en comparaison avec des modèles de régression et des réseaux de neurones.

2.4 Discussion

Nous nous sommes concentrés dans cette revue sur les modèles statistiques pouvant servir de base à la création d'indices et de cartes des prix immobiliers, nous n'avons donc pas évoqué tout un pan de la littérature, relative par exemple aux liens entre marché immobilier et ségrégation socio-spatiale [24, 39]. En effet, cette thèse, réalisée dans le cadre CIFRE au sein de l'entreprise MeilleursAgents, se plaçait d'emblée dans un horizon d'opérationnalité et en lien avec les méthodes déjà existantes chez MeilleursAgents. Il a donc été d'abord choisi d'explorer une « augmentation » des méthodes classiques : GWR et krigeage augmentés d'information issue d'un traitement algorithmique de données socio-économiques. Puis, au fil du travail, l'idée a germé d'un modèle entièrement nouveau mais beaucoup plus simple, prenant la ville elle-même, en tant que réseau de logements, comme un réseau de neurones apprenant son marché immobilier (différant en ceci d'une analyse de données réalisée par un réseau de neurones « quelconque »).

Parallèlement, dans le cadre de cette thèse, une nouvelle méthode de création d'indices des prix a été développée et mise en production chez MeilleursAgents en utilisant une notion de proximité socio-économique, et est opérationnelle depuis septembre 2022.

Chapitre 3

Données & protocole

Nous présentons ici d'abord les données qui seront utilisées au long du manuscrit. Elles se divisent en deux catégories principales : les données de transactions immobilières et les données sur la population. Puis nous présentons dans ses grandes lignes un protocole d'estimation, de test et d'optimisation qui sera employé à plusieurs reprises par la suite.

3.1 Données immobilières, issues de MeilleursAgents

MeilleursAgents (MA) est une entreprise créée en 2008, résultat de la rencontre entre des entrepreneurs et des chercheurs menant des travaux académiques sur la théorie des ventes répétées au sein de l'Université Paris-Dauphine. Ils partagent un constat commun, le marché des transactions immobilières repose sur un ensemble de connaissances et d'échanges essentiellement informels, non structurés et peu quantifiés. La plateforme de MA est destinée à mettre en relation les particuliers et les agences immobilières, grâce à différents outils dont les trois principaux sont :

- des indices d'évolution des prix de l'immobilier, disponibles partout en France à différentes granularités géographiques,
- une carte des prix précise et accessible à plusieurs échelles, en allant de l'échelle départementale à celle d'une adresse pour les grandes métropoles françaises,
- un outil d'estimation qui permet aux particuliers de faire une première estimation rapide et fiable de la valeur de leur bien en renseignant un certain nombre de ses caractéristiques.

Les données des transactions immobilières proviennent des bases de données de MA. Elles sont insérées manuellement par les agences partenaires, et de ce fait sont souvent remplies de manière incomplète. Nos cas d'étude utiliseront des transactions d'appartements localisés au niveau de la parcelle et vendus à une date précise (le mois et l'année de la transaction sont connus) en Île-de-France. Les variables descriptives des biens sont réunies dans le tableau 3.1.

TABLE 3.1 – Variables descriptives d'un appartement

Label	Variable
room_count	nombre de pièces
bathroom_count	nombre de salles de bain
secondary_room_count	nombre de chambres de bonne
floor	étage de l'appartement
area	surface de l'appartement (en m^2)
price	prix net vendeur auquel l'appartement a été vendu
pm^2	prix au mètre carré ($= \frac{price}{area}$)
parking_count	nombre de parkings associés à l'appartement
ascenseur	présence ou absence d'un ascenseur
balcon	présence ou absence d'un balcon
cave	présence ou absence d'une cave

Les données sont filtrées pour détecter les observations aberrantes. Selon le modèle utilisé, il est nécessaire d'ajouter une étape d'actualisation et/ou de normalisation au prétraitement de la donnée. L'étape d'actualisation consiste à mettre à jour le prix des transactions à la date souhaité en les ajustant grâce à l'indice des prix de l'immobilier. L'étape de normalisation permet quant à elle de ramener tous les appartements à un bien de référence afin de pouvoir les comparer entre eux en utilisant les coefficients produits par une régression hédonique.

3.1.1 Filtrage des données aberrantes

Pour notre étude, nous filtrons les données en fonctions des règles suivantes :

- room_count $\in \{1, 2, ..., , 15\}$
- bathroom_count $\in \{0, 1, 2, 3\}$
- secondary_room_count $\in \{0, 1, 2, 3, NA\}$
- parking_count $\in \{0, 1, 2, NA\}$
- floor_count $>= 0$

- area $\in [8, 500]$
- price > 0
- $\text{pm}^2 < 50000$
- $\text{pm}^2 > 0.25 \times \text{median}(\text{pm}^2)$

Les valeurs manquantes pour les variables *ascenseur*, *balcon*, *cave* sont arbitrairement remplacée par 0 (absence de ces caractéristiques).

3.1.2 Actualisation par l'indice des prix de l'immobilier

Si nécessaire, les données sont ajustées temporellement afin de tenir compte de la variation des prix au fil du temps : on ramène le prix de chaque transaction antérieure au mois précédent immédiatement celui où l'on veut faire une prédiction en l'ajustant à travers l'évolution d'un indice des prix. En l'occurrence, on calcule à partir de l'indice des prix $I(t)$ produit au sein de MeilleursAgents (et basé sur les prix observés) un coefficient d'évolution $\delta(t - t_a) = \frac{I(t)}{I(t_a)}$ entre l'instant t_a d'une transaction a et l'instant t où l'on veut l'actualiser : le prix actualisé sera simplement le prix observé multiplié par $\delta(t - t_a)$.

3.1.3 Normalisation des biens

Cette étape permet de ramener tous les biens à un même bien standard afin de produire des prix comparables entre eux. Pour ce faire, nous effectuons une régression hédonique où le bien de référence est un appartement de deux pièces et de 40 m^2, avec une salle de bain, au deuxième étage sans ascenseur et sans cave, parking ni balcon. Ceci est en effet le bien le plus représenté pour le principal cas de notre étude, à savoir Paris (voir 3.1).

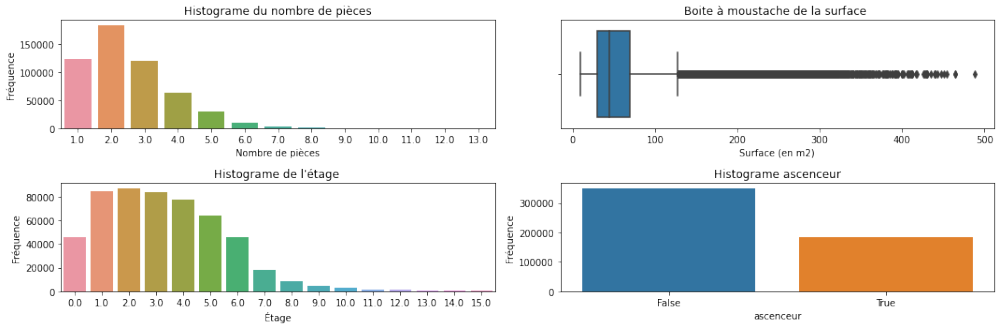


FIGURE 3.1 – Statistiques descriptives des appartements parisiens

Les variables utilisées pour la régression sont dérivées de celles présentées dans le tableau 3.1, en transformant les variables catégorielles en variables binaires pour chacune de leurs modalités (*dummification*). Nous obtenons ainsi 18 variables binaires et une variable numérique pour la superficie (sous la forme du logarithme de la superficie en m^2 divisée par 40), regroupées dans le tableau 3.2.

TABLE 3.2 – Variables descriptives d'un appartement

Label	Variable
$ROOM_1$	1 pièce
$ROOM_3$	3 pièces
$ROOM_4$	4 pièces
$ROOM_{5+}$	5 pièces ou plus
$FLOOR_1$	1er étage
$FLOOR_3$	3ème étage
$FLOOR_4$	4ème étage
$FLOOR_5$	5ème étage
$FLOOR_{6+}$	6ème étage ou plus
$BATHROOM_0$	0 salle de bain
$BATHROOM_2$	2 salles de bain
$BATHROOM_{3+}$	3 salles de bain ou plus
$PARKING_1$	1 parking
$PARKING_{2+}$	2 parkings ou plus
SEC_ROOM_0	une chambre de bonne ou plus
ELEVATOR	ascenseur
BALCONY	balcon
CELLAR	cave
LOG_AREA_NORM	$\log(\frac{area}{40})$

Nous repartons de l'équation de régression hédonique présentée au chapitre 1, équation 1.4, avec P_i le prix au m^2 de la transaction i :

$$\log(P_i) = \beta_0 + \sum_{k \in C} \beta_k X_k + \sum_{k \in S} \beta_k X_k + \sum_{k \in T} \beta_k X_k + \varepsilon_i \tag{3.1}$$

L'ensemble C correspond aux indices des 19 variables caractéristiques du bien (tableau 3.2), dont 18 variables binaires et une variable numérique (le logarithme de la surface). L'ensemble S regroupe les indices des variables spatiales, définies à partir d'une carte géographique : à Paris, une variable indicatrice pour chaque arrondissement ; hors Paris, une variable indicatrice pour chacune des 5 plus grandes villes

du département (en travaillant département par département). Enfin, l'ensemble T correspond aux variables temporelles : une variable indicatrice pour chaque année.

Nous souhaitons ici nous concentrer sur les dynamiques spatio-temporelles, il convient donc de neutraliser la part du prix relative aux caractéristiques — autrement dit nous allons travailler sur un prix normalisé qui est le prix d'un bien standard (appartement de deux pièces et de 40 m^2, avec une salle de bain, au deuxième étage sans ascenseur et sans cave, parking ni balcon) selon le quartier et l'année. Après avoir déterminé les coefficients β de la régression hédonique 3.1 grâce à l'ensemble des transactions des années $n-2$ et $n-1$, nous normalisons les prix des transactions observées dans les années n et $n+1$ à travers

$$P_i^{\text{norm}} = P_i \times \prod_{k \in F} \exp\left(-\beta_k X_k\right) \tag{3.2}$$

Le choix d'une fenêtre glissante de deux ans est arbitraire, fondé sur l'idée que les coefficients associés aux caractéristiques (i.e. la surcote ou la décote liée à telle ou telle caractéristique d'un logement) sont assez fondamentaux et restent relativement stables sur des périodes de quelques années.

3.2 Données sur la population, issues de l'INSEE

Les données socio-économiques seront utilisées ici comme variables de clustering pour établir des typologies de quartiers. Elles proviennent des bases de données publiques de l'INSEE (Institut national de la statistique et des études économiques) pour l'année 2010. Afin d'exploiter de façon optimale les résultats de l'analyse spatiale, nous décidons de travailler avec les données localisées au carreau, qui est le niveau de granularité le plus fin directement accessible sur le site de l'INSEE [31].

3.2.1 Présentation de la donnée carroyée

Le carroyage consiste en un découpage du territoire français en carreaux de 200 x 200 mètres, les données sont donc fournies sous forme de grille recouvrant entièrement le territoire. Afin de garantir la confidentialité relative aux données des

ménages, aucune information statistique (à l'exception du nombre total d'individus) n'est diffusée sur des carreaux de moins de 11 ménages. Les carreaux qui ne respectent pas cette condition sont regroupés en rectangles de taille croissante jusqu'à ce que la règle de confidentialité soit respectée. Ils vont de 1 carreau en zone dense à 3000 carreaux en zone très peu dense.

D'après la définition de l'INSEE, un ménage est défini comme l'ensemble des occupants d'une même unité d'habitation. Une unité de consommation, désignée par « u.c », est une abstraction par laquelle il est possible de comparer le niveau de vie des ménages de tailles différentes en attribuant à chaque personne du ménage un coefficient. Par exemple, le premier adulte du ménage compte pour une unité de consommation, les personnes de 14 ans ou plus comptent pour 0.5 u.c et les enfants de moins de 13 ans pour 0.3 u.c.
L'ensemble des variables fournies par l'INSEE au niveau du carreau est décrit dans le tableau 3.3.

TABLE 3.3 – Variables socio-économiques provenant de l'INSEE

Label	Variable
MEN	Nombre de ménages résidant dans le carreau
MEN_SURF	Surface cumulée des résidences principales, en mètres carrés
MEN_COLL	Nombre total de ménages en logement collectif
MEN_5IND	Nombre total de ménages de 5 personnes et plus
MEN_1IND	Nombre total de ménages d'une personne
MEN_PROP	Nombre total de ménages propriétaires
MEN_BASR	Nombre total de ménages dont le revenu fiscal par unité de consommation est en dessous du seuil de bas revenu
IND_R	Nombre total d'individus résidant dans le carreau
IND_SRF	Somme des revenus fiscaux par unité de consommation winsorisés des individus
IND_AGE1	Nombre total d'individus âgés de 0 à 3 ans
IND_AGE2	Nombre total d'individus âgés de 4 à 5 ans
IND_AGE3	Nombre total d'individus âgés de 6 à 10 ans
IND_AGE4	Nombre total d'individus âgés de 11 à 14 ans
IND_AGE5	Nombre total d'individus âgés de 15 à 17 ans
IND_AGE6	Nombre total d'individus âgés de 25 ans et plus
IND_AGE7	Nombre total d'individus âgés de 65 ans et plus
IND_AGE8	Nombre total d'individus âgés de 75 ans et plus

L'intérêt d'utiliser cette donnée est que les carreaux permettent de diffuser de

l'information statistique à un niveau faiblement agrégé et sont stables au cours du temps. D'autres données sont disponibles au niveau de l'IRIS (Ilots Regroupés pour l'Information Statistique), seulement la dimension et le tracé des IRIS varient au cours du temps, et ne contiennent pas non plus une population constante (en général de 1800 à 5000 habitants).

Néanmoins, l'utilisation de la donnée carroyée reste limitée. En effet, en ne tenant pas compte des limites administratives habituelles (rue, boulevard, périphérique, etc.), elle ne révèle pas l'hétérogénéité qui peut exister d'une rue à l'autre. De plus, un même carreau peut présenter deux populations aux caractéristiques très différentes en zone dense.

3.2.2 Traitement de la donnée

Nous choisissons de projeter les données carroyées sur un niveau géographique plus fin qui prend en compte les frontières géographiques, celui du bloc (polygone délimité par l'intersection des rues), en pondérant les valeurs des caractéristiques socio-économiques par la surface d'intersection entre le carreau et le bloc. Ainsi, la valeur d'une variable dans un bloc est la moyenne pondérée de ses valeurs sur les carreaux intersectant le bloc.
Plus précisément, en notant n le nombre total de blocs et n_c le nombre total de carreaux dans une ville, pour chaque bloc j on calcule sa représentativité dans les carreaux c qu'il intersecte. On note $A(.)$ l'aire et on définit le ratio $r(j, c)$ par :

$$r(j, c) = \frac{A(j \cap c)}{A(c)} \tag{3.3}$$

Par souci de simplicité, on utilise j pour décrire l'indice du bloc et le bloc lui-même (pareillement pour le carreau c).
Ainsi pour une variable q représentant une quantité, la valeur de la variable pour le bloc j sera donnée par :

$$q_j = \sum_{c=1}^{n_c} r(j, c) q_c. \tag{3.4}$$

où q_c est la valeur de la variable q dans le carreau c.

Après avoir projeté les variables au carreau sur le niveau du bloc, celles-ci sont

transformées afin d'être exploitables pour la construction des clusters en calculant des ratios. Les variables finales avec lesquelles nous travaillons pour la classification SOM sont décrites dans le tableau 3.4.

TABLE 3.4 – Variables finales utilisées pour le clustering

Label	Variable
MEN_SURF_MOY	Surface moyenne de la résidence principale, en mètres carrés
MEN_COLL_PCT	Pourcentage de ménages en logement collectif
MEN_5IND_PCT	Pourcentage de ménages de 5 personnes et plus
MEN_1IND_PCT	Pourcentage de ménages d'une personne
MEN_PROP_PCT	Pourcentage de ménages propriétaires
MEN_BASR_PCT	Pourcentage de ménages dont le revenu fiscal par unité de consommation est en dessous du seuil de bas revenu
IND_SRF_MOY	Revenu fiscal moyen par unité de consommation
IND_18_25_PCT	Pourcentage d'individus âgés de 18 à 25 ans
IND_65P_PCT	Pourcentage d'individus âgés de 65 ans et plus
DENSITE_POP	Densité de population (en hab. par km carré)

Parmi les variables INSEE disponibles, le choix a été fait de façon subjective, on pourrait aussi utiliser un outil de sélection de variables afin d'enlever tout biais humain, mais il n'est pas évident que cela soit pertinent ici. Surtout, nous pourrions ajouter des variables décrivant les biens et services disponibles dans les quartiers, mais la contrainte opérationnelle impose de rester parcimonieux dans les données supplémentaires et d'utiliser des données pertinentes sur l'ensemble du territoire français au moins.

3.3 Protocole

Nous décrivons ici un protocole qui sera commun aux deux chapitres suivants, et légèrement modifié dans le chapitre 6.

3.3.1 Estimation et prédiction

Considérons un estimateur $\hat{P}^{(\boldsymbol{\sigma})}(u)$ du prix d'un bien à la localisation u avec des caractéristiques exprimées par des valeurs des variables du tableau 3.2. L'estimateur dépend d'un paramètre $\boldsymbol{\sigma}$ potentiellement multidimensionnel. Par exemple, si l'estimateur provient d'une GWR réalisée avec un noyau gaussien, il dépend de la

variance σ^2 du noyau gaussien. Si l'estimateur provient d'une régression hédonique, il est indépendant de u, et ne dépend éventuellement d'aucun paramètre – sauf si l'on décide par exemple d'introduire des variables indicatrices correspondant à un découpage spatial et de paramétrer ce découpage.

Pour chaque choix de $\boldsymbol{\sigma}$, si l'on dispose d'un jeu de données antérieures à un instant t (indiqué en année-mois) on peut estimer $\hat{P}^{(\boldsymbol{\sigma})}(u)$ et utiliser cette valeur pour prédire un prix à l'instant t à la localisation u. Dans la suite on appellera $D_{\text{train}}(t)$ ou simplement D_{train} le jeu de données utilisé pour fournir une prédiction à l'instant t. Généralement, il s'agira des transactions ayant eu lieu dans la même ville pendant les 4 années antérieures à t. Ces données sont ajustées temporellement afin de tenir compte de la variation des prix au fil des quatre années : on ramène le prix de chaque transaction au mois précédent immédiatement la prédiction en l'ajustant à travers l'évolution d'un indice des prix comme indiqué au paragraphe 3.1.2.

$D_{\text{train}}(t)$ est donc constitué des transactions des 4 dernières années, actualisées via l'indice des prix. $\hat{\beta}_0(u)$ est estimé sur $D_{\text{train}}(t)$ et fournit des prédictions de prix en chaque localisation u. Les transactions observées pendant le mois t constituent ensuite un ensemble de test, $D_{\text{test}}(t)$, sur lequel on peut mesurer les erreurs commises en mesurant l'erreur relative pour chaque transaction $a \in D_{\text{test}}(t)$ (en pourcentage) :

$$E_a^{(\sigma)} = \frac{|\hat{P}^{(\boldsymbol{\sigma})}(u_a) - P_a|}{P_a} \times 100, \tag{3.5}$$

où $\hat{P}^{(\boldsymbol{\sigma})}(u_a)$ est le prix prédit à la localisation de a à partir du modèle estimé sur $D_{\text{train}}(t)$, et P_a est le prix observé. Les prix prédits, et donc l'erreur commise, dépendent du choix de $\boldsymbol{\sigma}$.

3.3.2 Optimisation des paramètres

Plutôt que de conserver un choix arbitraire des paramètres $\boldsymbol{\sigma}$, on les estime à travers un critère de minimisation de la médiane des erreurs de prédiction commises sur une période donnée, que nous noterons T_{opt}.

Ainsi, pour chaque $t \in T_{\text{opt}}$ et chaque valeur de $\boldsymbol{\sigma}$ on réalise estimation et prédiction tel que décrit dans la section précédente, puis on calcule les erreurs relatives

commises. On dispose alors d'un ensemble

$$\bigcup_{t \in T_{\text{opt}}} \left\{ E_a^{(\sigma)}, a \in D_{\text{test}}(t) \right\}$$

qui sont les valeurs des erreurs sur la période T_{opt} pour un choix donné de $\boldsymbol{\sigma}$. On détermine la médiane de cet ensemble :

$$m_{\boldsymbol{\sigma}} = \text{median} \left(\bigcup_{t \in T_{\text{opt}}} \left\{ E_a^{(\sigma)}, a \in D_{\text{test}}(t) \right\} \right). \tag{3.6}$$

Ne pouvant explorer une infinité non dénombrable de valeurs de $\boldsymbol{\sigma}$, on choisit une région de valeurs possibles que l'on explore avec un certain pas : par exemple, si $\boldsymbol{\sigma}$ est un scalaire et raisonnablement compris entre 10 et 100, on peut explorer l'intervalle $[10, 100]$ par pas de 1. On déterminera donc m_{10}, m_{11}, m_{12}, etc. Cette exploration s'appelle *grid search* en anglais, puisqu'on discrétise l'espace à explorer au moyen d'une grille (d'un réseau) de maille fixée. On note $\boldsymbol{\Sigma}$ la grille, c'est à dire l'ensemble fini de valeurs à explorer pour $\boldsymbol{\sigma}$. La valeur qui sera retenue est alors définie par :

$$\boldsymbol{\sigma}^* \; = \; \text{argmin}_{\boldsymbol{\sigma} \in \boldsymbol{\Sigma}} \, m_{\boldsymbol{\sigma}}. \tag{3.7}$$

Dans la suite, aux chapitres 4 et 5 notamment, nous prendrons généralement une période d'optimisation de 3 ans, T_{opt}, courant de 2016-01 à 2018-12. Les comparaisons seront ensuite réalisées sur les prédictions faites pour la période T_{val} allant de 2019-01 à 2021-01.

Dans le chapitre 6, les neurones du réseau seront initialisés à 2016-01 avec une profondeur temporelle de 4 ans, puis le réseau se mettra à jour au fur et à mesure (chaque mois) avec les données de 2016-01 à 2021-01, avec comme pour les autres modèles une période d'optimisation des paramètres du réseau entre 2016-01 et 2018-12, à ceci près que nous serons amenés à chercher d'autres techniques d'optimisation qu'un *grid search* pour des raisons de temps de calcul.

Chapitre 4

GWR x SOM

Nous explorons ici une première piste d'amélioration des méthodes existantes en « augmentant » la GWR d'informations tirées d'une typologie de quartier réalisée à l'aide de l'algorithme SOM sur les données socio-économiques de l'INSEE (tableau 3.4). SOM donne une notion de proximité entre les clusters et fournit ainsi un degré de similarité multiéchelle. L'information de clustering est ensuite utilisée pour modifier les poids de la GWR, comme nous l'expliquons dans la section 4.1. Nous appliquons ensuite le nouveau modèle (GWRxSOM) sur deux villes relativement différentes : Paris et Les Lilas.

4.1 Pondération par la proximité socio-économique

Nous repartons de la GWR présentée dans le chapitre 1, en appliquant le protocole présenté dans la chapitre 3.

4.1.1 GWR seule

L'ensemble des $p = 19$ variables descriptives des caractéristiques d'un logement sont celles du tableau 3.2. On choisit un noyau gaussien pour la GWR, avec des poids donnés donc par (cf. équation 1.7) :

$$W(\mathbf{v} - \mathbf{u}) = \exp\left(-\frac{||\mathbf{v} - \mathbf{u}||^2}{2\,\sigma^2}\right). \tag{4.1}$$

Comme décrit dans le protocole présenté au chapitre 3, chaque mois t et pour chaque localisation dans la ville un modèle GWR (cf. équation 1.8) est entrainé sur les données d'entrainement D_{train} constitué de l'ensemble des observations des 4 années (strictement) antérieures à t. L'ensemble des observations du mois t est utilisé pour former l'ensemble de test sur lequel on évalue les erreurs relatives. Et ainsi de suite, mois après mois pour t allant de 2016-01 à 2021-01, les trois premières années (de 2016-01 à 2018-12) étant en outre utilisées pour déterminer la valeur optimale du paramètre σ apparaissant dans le noyau gaussien de la GWR (cf. chap. 3). Un exemple de profil d'erreur médiane en fonction de σ est présenté sur la figure 4.1.

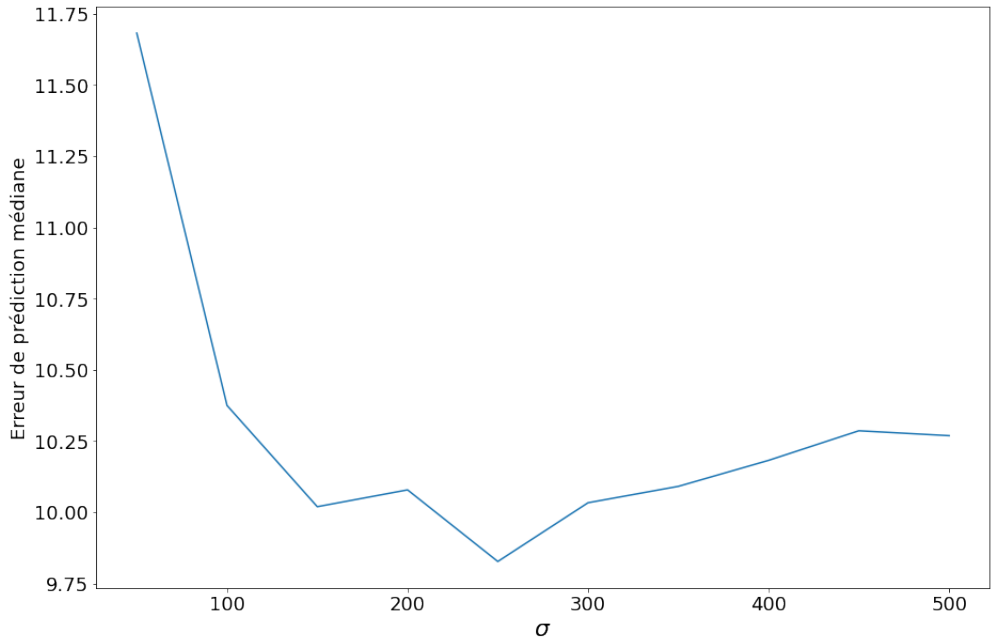


FIGURE 4.1 – Erreurs de prédiction médianes sur la période T_{opt} =[2016-01,2018-12] lors d'un *grid search* pour le rayon caractéristique σ de la GWR appliquée aux Lilas.

La section suivante décrit la manière dont on augmente la GWR par l'algorithme SOM.

4.1.2 GWR x SOM

L'information liée au clustering peut être directement intégrée à la GWR d'une manière très simple en intervenant multiplicativement sur les poids. Nous définissons dans un premier temps les poids SOM.

Pour chaque transaction a, on cherche le bloc j dont elle fait partie dans la ville, désigné par $j(a)$ et le vecteur y associé à a est noté $y_{j(a)}$. À partir de la distance SOM (équation 1.19), on définit les poids SOM pour une localisation u par :

$$w_i^{SOM}(u) = \exp\left(-\frac{d_{SOM}(y_{j(i)}, y_{j(u)})^2}{2\gamma^2}\right), \tag{4.2}$$

avec γ un paramètre positif jouant le rôle d'une « distance » caractéristique entre les clusters SOM.

Pour notre modèle, pour chaque transaction u, nous prenons en compte uniquement les transactions a telles que $y_{j(a)}$ et $y_{j(u)}$ appartiennent à des clusters voisins sur la carte SOM ; voisins au sens d'un rayon r_{SOM} donné. Ce rayon dépend de la taille de la carte SOM, par exemple dans cette étude pour une carte 3x3, nous prenons un rayon sur la carte de 1 ; pour des tailles de cartes SOM plus élevées, 15x15 par exemple, alors nous choisissons un rayon de 3 [1].
À une localisation u, nous disposons également des poids standard de la GWR, $w_i^{GEO}(u) = \exp\left(-\frac{||\mathbf{u_i}-\mathbf{u}||^2}{2\,\sigma^2}\right)$, avec u_i la localisation de la transaction i. Une manière simple de combiner ceux-ci avec les nouveaux poids SOM est de les multiplier :

$$w_i^F(u) = w_i^{GEO}(u) \times w_i^{SOM}(u), \tag{4.3}$$

si bien qu'une faible proximité géographique peut être compensée par une grande proximité socio-économique, et vice-versa. En outre, une grande proximité géographique combinée à une grande proximité socio-économique donnera un poids maximal. En effet, si on considère un point i se situant exactement au même endroit que le point courant u, alors $w_i^{GEO}(u) = 1$, et si i et u appartiennent au même cluster SOM, alors $w_i^{SOM}(u) = 1$; dans ce cas $w_i^F(u) = 1$. Les poids combinés décroissent vers 0 quand i est loin de u du point de vue géographique et/ou du point de vue socio-économique.
La fonction à minimiser pour estimer le modèle GWRxSOM est alors simplement :

$$\mathcal{E}(u) = \sum_{i=1}^{N} w_i^F(u)\varepsilon_i^2. \tag{4.4}$$

1. En ne fixant pas de rayon et en prenant toutes les classes en considération, les performances étaient moins bonnes.

Nous appliquerons avec cette nouvelle méthode le même protocole que pour la GWR seule (protocole général présenté au chapitre 3). En particulier, les paramètres σ et γ apparaissant dans les poids w_i^{GEO} et w_i^{SOM} sont choisis conjointement par *grid search* en minimisant l'erreur médiane de prédiction sur la période T_{opt}=[2016-01,2018-12].

La section suivante compare les résultats après application du nouveau modèle et ceux après une simple GWR.

4.2 Applications sur deux villes réelles

Nous travaillons sur deux villes avec un contexte de données très différent : Paris, où s'effectuent environ 30 000 transactions par an [2], et Les Lilas, où s'effectuent environ 171 transactions d'appartements par an. Les Lilas est une ville périphérique à l'est de Paris où le parc est mixte avec une majorité d'appartements en centre-ville et autour du métro. Pour cette étude nous ne considérons pas les maisons individuelles, seulement les appartements.

TABLE 4.1 – Description du parc

	Paris	Les Lilas
Population	2220445	22762
Densité de population (hab./km^2)	21010	18759
Nombre de logements (en millier)	1330.026	11.133
% d'appartements	99	89
Nombre de parcelles	77218	1880
Nombre de parcelles résidentielles	66295	1644
Nombre de carreaux	2074	46
Nombre de blocs	7613	67

2. Il s'agit d'une moyenne entre 2012-01 et 2020-12.

TABLE 4.3 – Statistiques descriptives des variables de clustering

	Paris			Les Lilas		
	Min	Median	Max	Min	Median	Max
MEN_SURF_MOY	21.94	54.9	142.9	48.8	56.4	77.2
MEN_COLL_PCT	48.9	98.4	100	48	91.2	99.7
MEN_5IND_PCT	0	5.1	44.4	3	7.1	16.4
MEN_1IND_PCT	11.5	50.1	79.2	20.6	42.6	51
MEN_PROP_PCT	20	36.7	69.8	20	43.9	78.3
MEN_BASR_PCT	1.5	15.8	60	7.9	18.3	29.4
IND_SRF_MOY	10960	23028	27844	16821	20771	23267
IND_18_25_PCT	0.3	7.4	23	6	7.6	10.4
IND_65P_PCT	0.8	15	45.9	8.5	12.5	22.4
DENSITE_POP	0.77	31875	88452	2259	20124	41618

TABLE 4.2 – Statistiques descriptives des caractéristiques des appartements (2020).

	Paris	Les Lilas
Nombre de pièces (fréquence max)	2	2
Étage (fréquence max)	1	1
Nombre de salles de bain (fréquence max)	1	1
Nombre de parkings (fréquence max)	0	0
% chambre de bonne	2	0
% ascenseur	42	46
% balcon	14	30
% cave	69	71
Surface médiane (en m^2)	45	48
Prix m^2 médian (en €)	11000	7193
Nombre d'observations	16999	166

4.2.1 SOM sur les données INSEE à Paris et aux Lilas

Nous appliquons l'algorithme SOM sur les données socio-économiques à Paris et aux Lilas. Les données socio-économiques utilisées devraient idéalement être les plus récentes possible, cependant au moment où nous avons commencé ce travail, seules les données de 2010 étaient disponibles au carreau, malheureusement.

Nous choisissons de travailler avec 225 clusters (carte SOM 15x15) pour Paris et 9 clusters (carte SOM 3x3) pour Les Lilas. Nous exécutons 100 itérations de

l'algorithme et choisissons l'itération qui minimise le critère d'inertie intra-classe :

$$I_{ic} = \frac{1}{n} \sum_{k=1}^{K} \sum_{i \in C_k} ||y_i - \overline{y}_k|| \tag{4.5}$$

Nous réalisons ensuite une CAH afin de définir 10 super-clusters sur Paris que l'on projette sur la carte de la ville (figure 4.2).

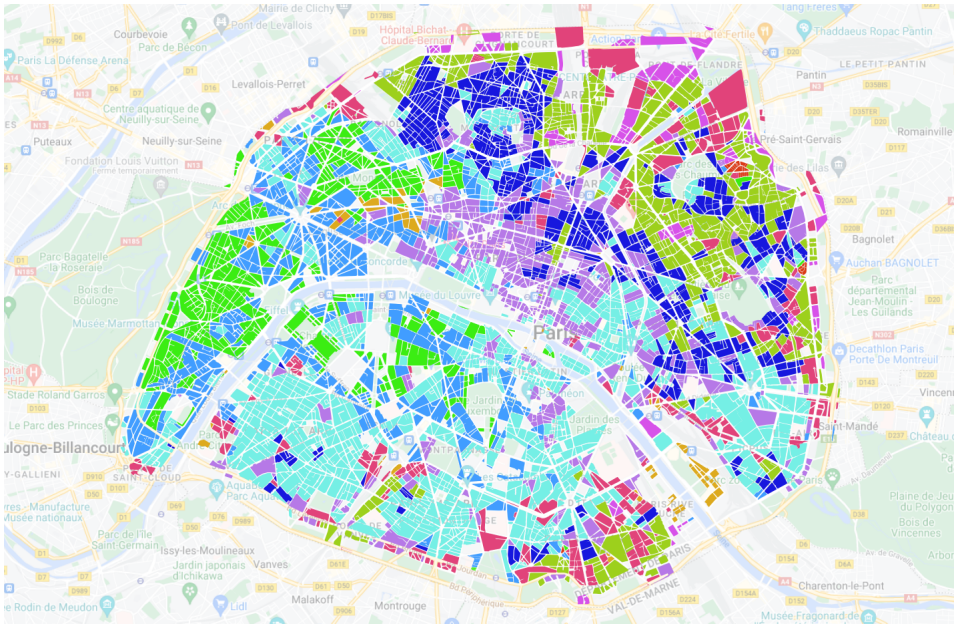


FIGURE 4.2 – Ville de Paris après avoir appliqué SOM au niveau du bloc (carte SOM 15x15 et 10 super-clusters).

Les quartiers très bourgeois de l'Ouest parisien et ceux du 7ème arrondissement (super-cluster vert) se distinguent d'autres quartiers aisés (super-cluster bleu clair). On remarque également des quartiers comme ceux à la fois commerçants, de bureaux et résidentiels entre les Halles et les gares Saint Lazare, du Nord et de l'Est (super-cluster violet), des quartiers gentrifiés autour de Montmartre dans le 18ème arrondissement, du parc des Buttes-Chaumont ou du Canal (alliant super-clusters bleu foncé et bleu clair).

La carte des Lilas après clusterisation par SOM est présentée quant à elle sur la figure 4.3.

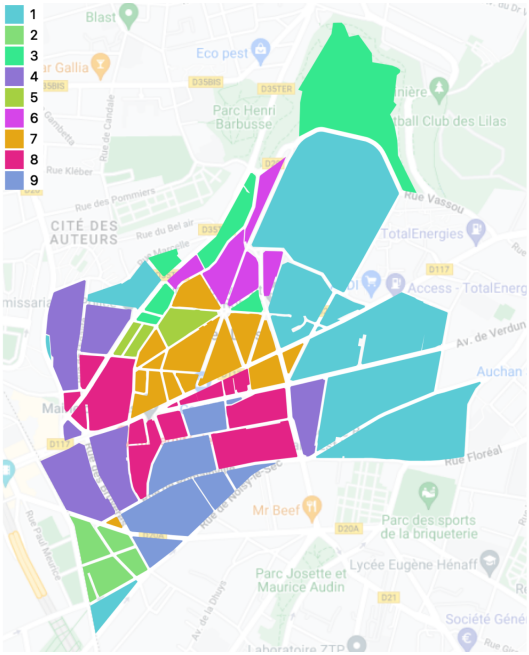


FIGURE 4.3 – Les Lilas après avoir appliqué l'algorithme SOM au niveau du bloc (carte SOM 3x3)

Une forme de quartier central ressort (cluster 7, orange) tandis que la périphérie se distingue entre l'ouest, plus près de Paris, et l'est, plus éloigné de Paris.

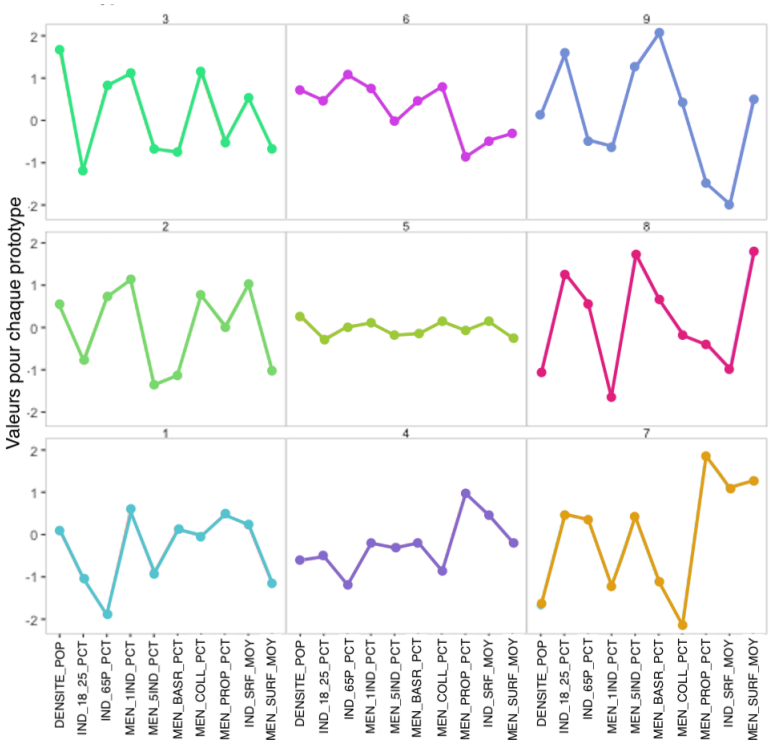


FIGURE 4.4 – Prototypes de la carte SOM pour Les Lilas. Dans chaque unité, on représente un vecteur q-dimensionnel ($q = 10$) dont les composantes sont les variables listées dans le tableau 3.3.

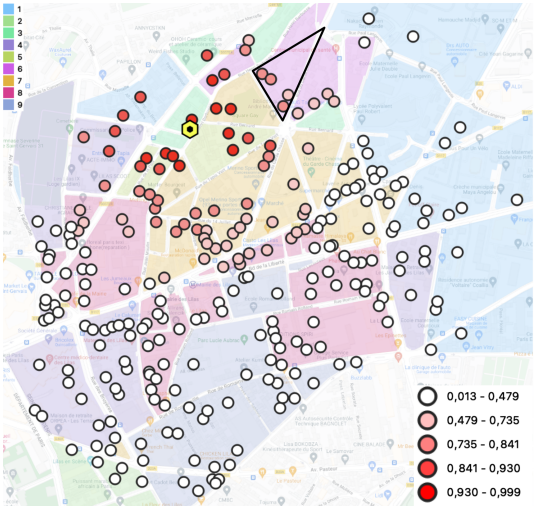
À défaut d'analyser les prototypes pour la ville de Paris (car trop nombreux), on représente ceux de la ville des Lilas dans la Figure 4.4, qui sont des vecteurs 10-dimensionels. On peut voir que les prototypes des unités 3 et 7 opposés sur la carte SOM sont très différents. En effet, la variable 1 qui représente la densité de population dans un bloc (désignée par « DENSITE_POP ») a un niveau élevé pour le prototype de l'unité 3, alors qu'elle a un niveau faible pour le prototype de l'unité 7. De la même manière, le niveau de la variable 2 qui représente le pourcentage d'individus âgés de 18 à 25 ans dans un bloc (désignée par «IND_18_25_PCT») est faible pour l'unité 3 et élevé pour l'unité 7. Et ainsi de suite pour les variables suivantes. Si on part de l'unité 1 à l'unité 9 en passant par l'unité 5, on remarque que le prototype de l'unité 5 (qui est au centre de la carte SOM) est relativement plat et donc, comparable à tous les prototypes de la carte.

Des tests sur la taille de la carte SOM pour la ville des Lilas ont été effectués : entre une taille 3x3 et une taille 4x4, c'est la première qui minimise l'erreur de prédiction. Aucun test sur la taille de la carte SOM n'a été effectué sur la ville

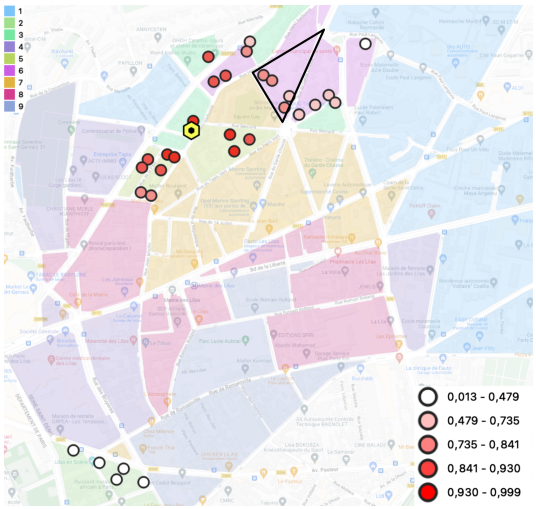
de Paris, car l'algorithme SOM est très long à exécuter. Cependant, le nombre de clusters a été choisi de manière à avoir un nombre de blocs par cluster satisfaisant (7613/(15×15)=33.8 si on suppose une répartition homogène). Le rayon de voisinage sur la carte SOM r_{SOM} est choisi sans *grid search* et vaut r_{SOM}=1 pour la ville Les Lilas, car prendre un rayon supérieur reviendrait à considérer tous les clusters sur la carte. À Paris nous choisissons r_{SOM}=3 pour être sur de couvrir suffisamment de blocs. Cela revient à considérer entre 16 et 49 voisins sur la carte SOM (selon la position du cluster), et donc 33.8x16=540.8 blocs en moyenne pour le «pire» des cas.

4.2.2 Modification des poids de la GWR

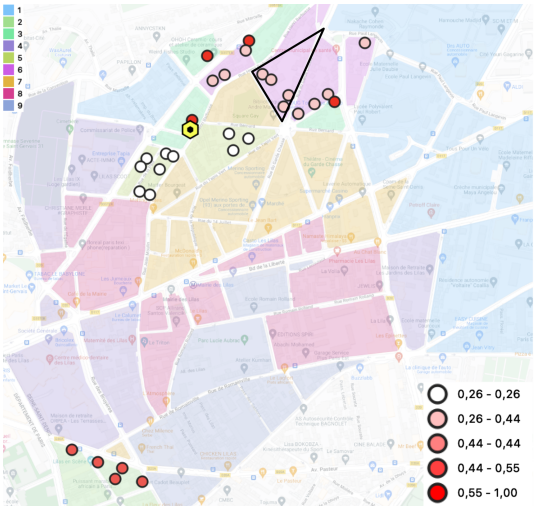
Un exemple de l'impact des poids SOM sur les poids géographique est montré figure 4.5 pour une localisation donnée aux Lilas.



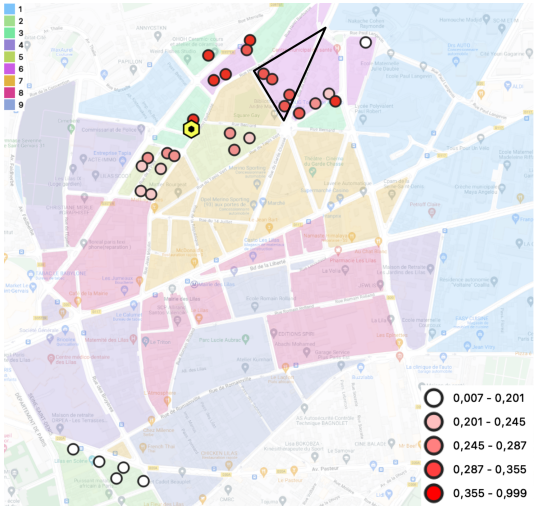
(a) Poids géographique pour la parcelle courante (hexagone jaune)



(b) Poids géographiques filtrés par les classes non voisines de la parcelle courante



(c) Poids SOM pour la parcelle courante



(d) Poids finaux pour la parcelle courante.

FIGURE 4.5 – Exemple de l'impact de SOM dans la définition des poids. Carte aux Lilas au 1er janvier 2021. L'hexagone jaune représente la parcelle pour laquelle on applique les poids et le triangle noir regroupe des poids intéressants à analyser.

Si on se penche sur les poids situés à l'intérieur du triangle noir, les transactions localisées dans cet endroit auront plus d'importance avec notre nouveau modèle (Figure 4.5 (d)) que si on avait pris en compte les poids géographiques seulement (Figure 4.5 (b)). En effet, le cluster auquel appartiennent les transactions du triangle noir (cluster 6) et celui de la parcelle courante (cluster 3) sont voisins sur la carte

SOM et donc présentent des caractéristiques similaires (Figure 4.5(c)).

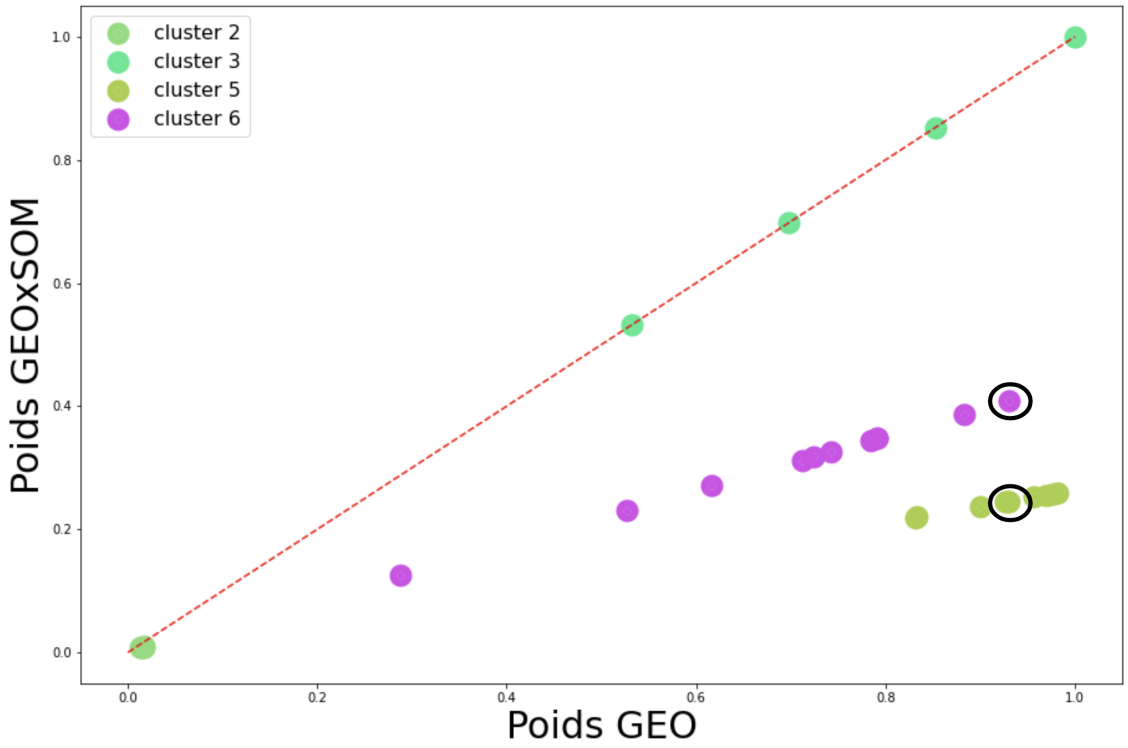
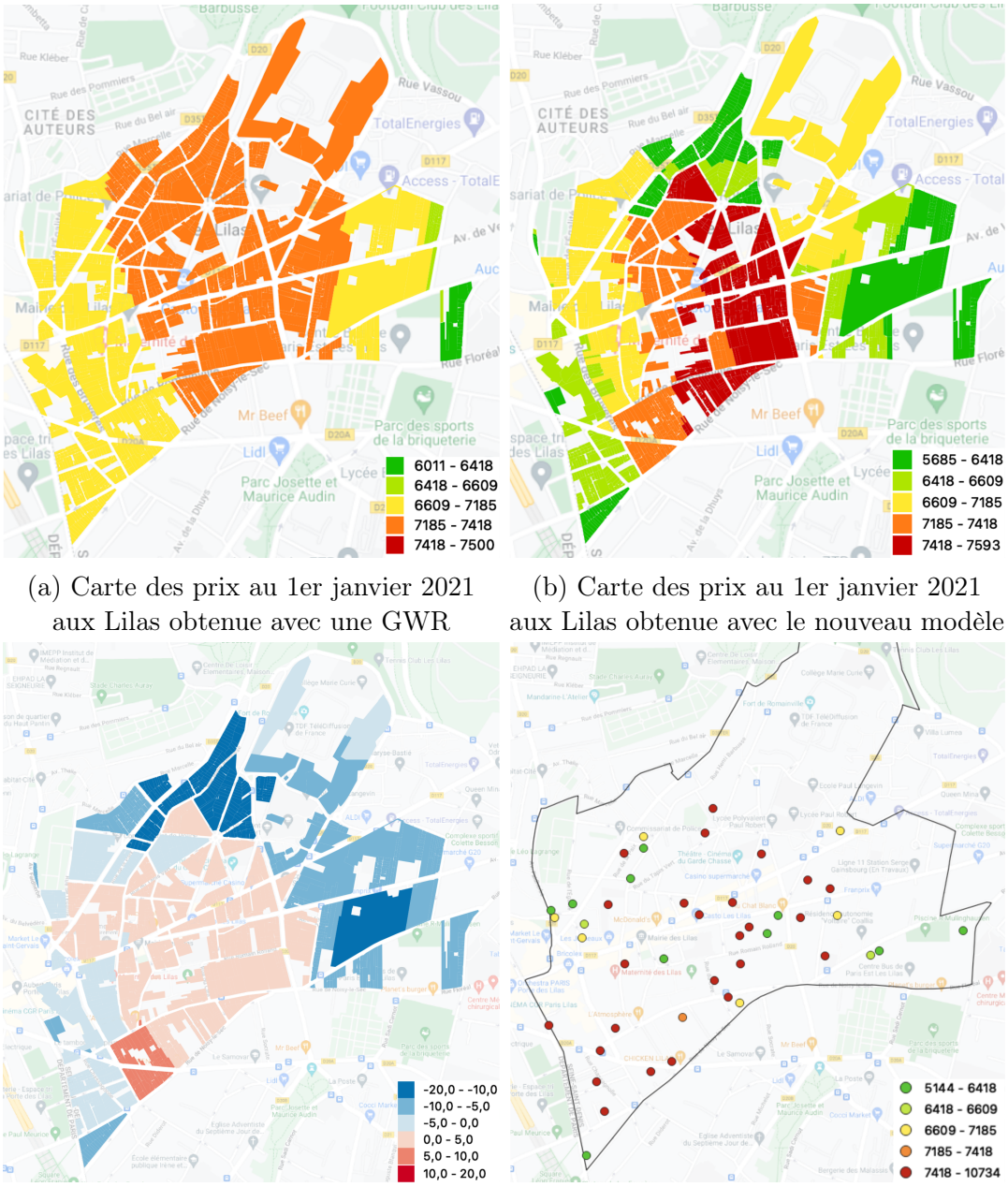


FIGURE 4.6 – Nuage de points des poids présentés dans les Figures 4.5 (b) et (d). Les poids géographiques sont en abscisse et les poids finaux en ordonné.

La Figure 4.6 représente les poids liés à la parcelle courante de la Figure 4.5 (hexagone jaune). En abscisse les poids géographiques seulement et en ordonnée les poids croisés. Pour une même distance (et donc pour des poids géographiques égaux) les nouveaux poids relatifs sont pénalisés en fonction de la dissimilitude des clusters liés aux transactions par rapport au cluster lié à la parcelle courante. Les transactions appartenant au cluster 3 ne voient pas leur poids changer, car la distance SOM est nulle, et donc les poids égaux à 1. Les poids du cluster 5 sont plus pénalisés que ceux du cluster 6 car moins similaires du point de vue socio-économique par rapport au cluster 3. En effet, les poids entourés d'un cercle noir ont une valeur initiale de 0.93 pour avoir une valeur finale de 0.4 et 0.2 respectivement.

Examinons maintenant les prédictions de prix produites selon le protocole général décrit au chapitre 3. La prédiction se fait au niveau de la parcelle notée u et le prix est celui d'un bien standard qui serait localisé en u, c'est à dire l'intercept $\hat{\beta}_0(u)$

obtenu dans le modèle 1.8. Les cartes des prix obtenus pour les différents modèles sur les deux villes de Paris et Les Lilas sont affichés sur les figures 4.7 et 4.8.



(a) Carte des prix au 1er janvier 2021 aux Lilas obtenue avec une GWR

(b) Carte des prix au 1er janvier 2021 aux Lilas obtenue avec le nouveau modèle

(c) Différence de prix en pourcentage entre les figures précédentes

(d) Prix au m^2 normalisé des observations du premier trimestre 2021

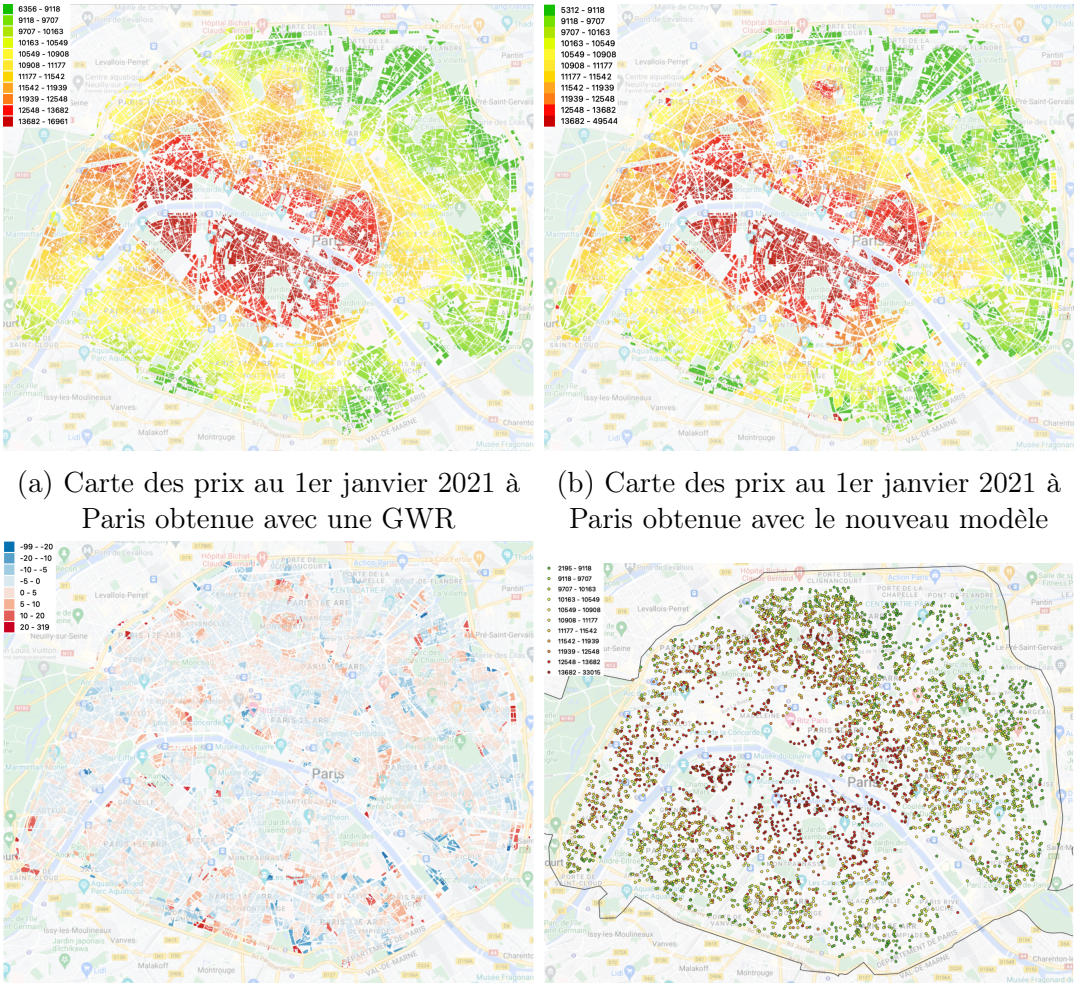
FIGURE 4.7 – Prix prédits au 1er janvier 2021 et observations au premier trimestre 2021, aux Lilas.

La carte des prix aux Lilas produite avec une GWR (figure 4.7 (a)) comporte

un gradient de prix de l'est vers l'ouest qui reste relativement homogène du nord au sud alors qu'aux Lilas il existe un réel effet centre-ville, du fait de la présence du métro et d'un bâti plus recherché près de la mairie. Sur la figure 4.7(b), on voit que l'information issue de SOM structure une spatialisation des prix de la ville différente, où un centre-ville et une périphérie se construisent plus clairement, correspondant mieux à la réalité du terrain (comme on peut s'en rendre compte également en l'occurrence en se rendant sur place).
La figure 4.7 (c) représente la différence entre les prix produits par la GWR seule (figure 4.7 (a)) et ceux produits avec notre nouvelle méthode (figure 4.7(b)). La différence de prix pour une parcelle i est définie par :

$$\Delta_i(t) = \frac{|P_i(t)^{\mathrm{GWRxSOM}} - P_i(t)^{\mathrm{GWR}}|}{P_i(t)^{\mathrm{GWR}}} \times 100, \tag{4.6}$$

où $P_i(t)^{\mathrm{GWRxSOM}}$ est le prix prédit par la nouvelle méthode en la parcelle i et $P_i(t)^{\mathrm{GWR}}$ est celui prédit par la GWR seule.



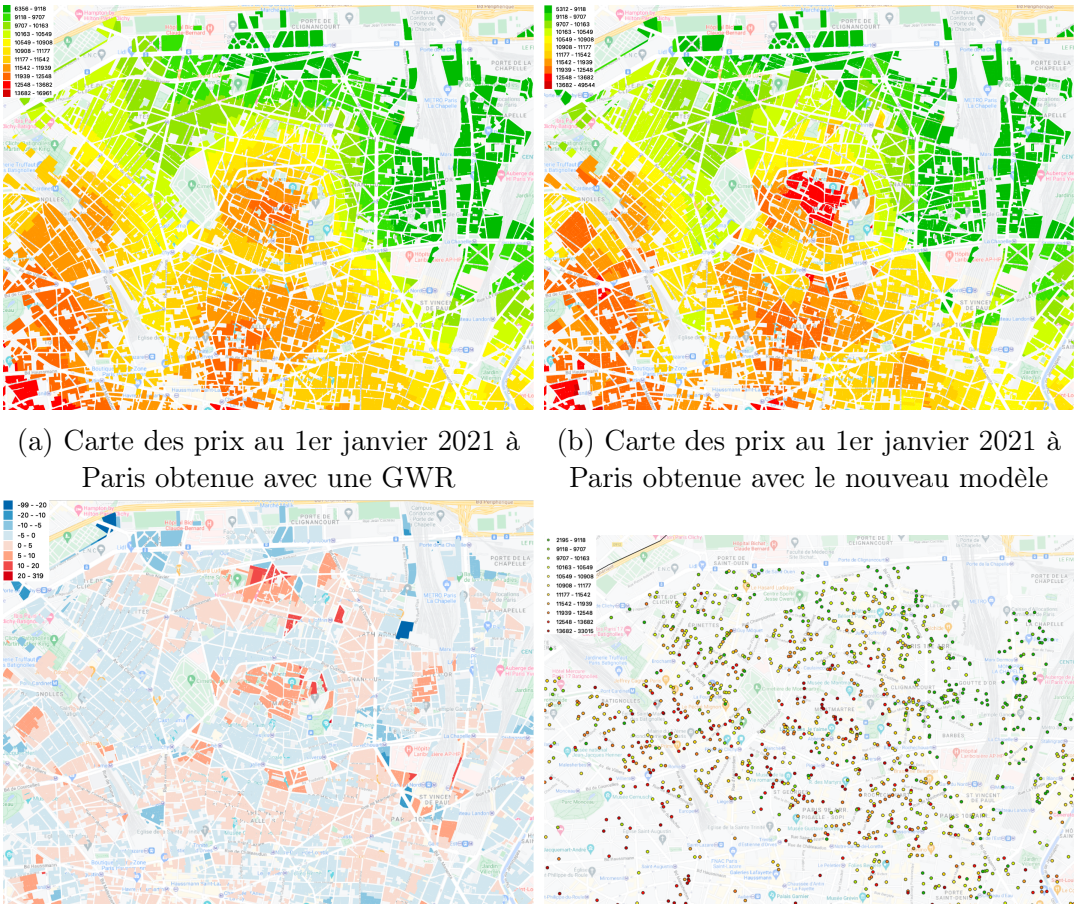
(a) Carte des prix au 1er janvier 2021 à Paris obtenue avec une GWR

(b) Carte des prix au 1er janvier 2021 à Paris obtenue avec le nouveau modèle

(c) Différence de prix en pourcentage entre les figures précédentes

(d) Prix au m^2 normalisé des observations du premier trimestre 2021

FIGURE 4.8 – Prix prédits au 1er janvier 2021 et observations au premier trimestre 2021, à Paris.



(a) Carte des prix au 1er janvier 2021 à Paris obtenue avec une GWR

(b) Carte des prix au 1er janvier 2021 à Paris obtenue avec le nouveau modèle

(c) Différence de prix en pourcentage entre les figures précédentes

(d) Prix au m^2 normalisé des observations du premier trimestre 2021

FIGURE 4.9 – Prix prédits au 1er janvier 2021 et observations au premier trimestre 2021 à Paris, zoom sur le quartier de Montmartre et ses alentours

À Paris le constat est plus nuancé, la structure globale des cartes de prix est semblable d'un modèle à l'autre (Figure 4.8(a) vs Figure 4.8(b)). Cependant, on voit bien l'effet de l'algorithme SOM (Figure4.2) à Montmartre (Figure 4.9). En effet, les blocs au sud de la rue Caulaincourt et au nord-ouest du Sacré-Coeur appartiennent à un cluster non contigu avec les autres blocs du voisinage. Les transactions observées dans ces blocs répondent donc à des mêmes dynamiques de prix que celles observées dans d'autres quartiers de Paris plus éloignés géographiquement, mais appartenant au même cluster où les prix sont plus élevés.

La représentation des prix par projection sur des cartes géographique permet

de se rendre compte de la dissimilarité spatiale globale, mais rend difficile l'analyse des différences dues à l'intégration de l'information de clustering. C'est pourquoi on représente les prix en nuage de points avec en abscisse les prix obtenus avec une GWR seule et en ordonnée les prix obtenus avec le nouveau modèle GWR+SOM. Chaque point correspond au prix d'une parcelle, l'appartenance à un cluster est représentée par sa couleur. Examinons cette représentation pour Les Lilas (figure 4.10).

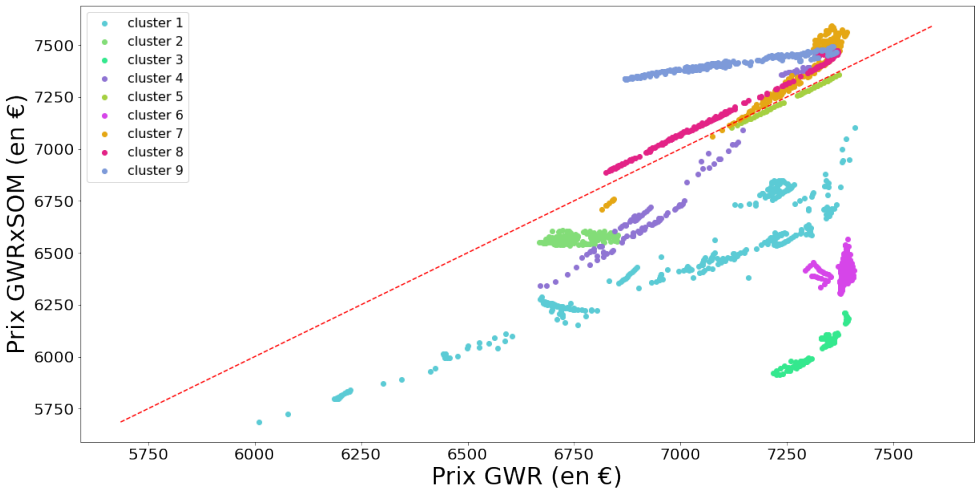
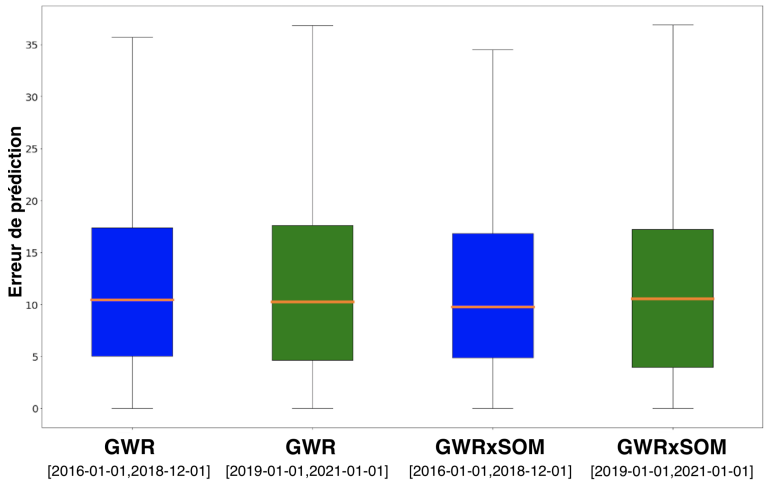


FIGURE 4.10 – Diagramme de dispersion des prix obtenus avec les modèles de GWR seule et de GWR+SOM au 1er janvier 2021 aux Lilas.

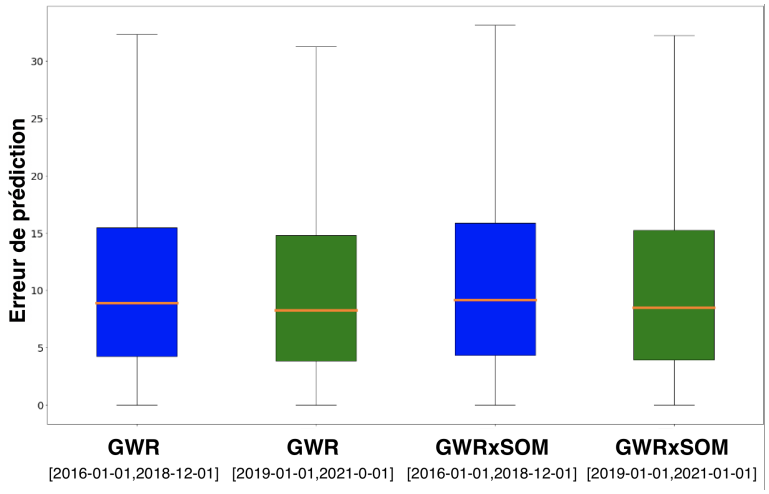
Si la variation de prix issus d'une GWR par rapport à ceux issus du nouveau modèle reste relativement faible, il est intéressant de remarquer les différences de variation par rapport à l'appartenance à un cluster. Les prix des parcelles localisées sur le cluster 3 diminuent de 17% en moyenne et cela est dû à une observation dont le prix de la transaction est parmi les plus bas de l'ensemble d'entrainement D_{train}. SOM tel qui est appliqué dans notre modèle a pour effet de donner plus d'importance pour des observations appartenant au même cluster, l'observation ayant un prix faible a une importance relative plus élevée et donc tire le prix prédit des parcelles appartenant au cluster 3 vers le bas. Il est donc important de bien filtrer les données en amont. Au contraire, les prix des parcelles appartenant au cluster 5 ont une variation quasi nulle en passant d'un prix GWR aux prix issu du nouveau modèle. En effet, le cluster 5 se situant au milieu de la carte SOM, son prototype est plat et est relativement comparable à tous les autres prototypes de la carte. Les distances du cluster 5 par rapport aux autres clusters sont quasiment identiques et cela a pour effet de ne pas modifier les poids relatifs.

4.2.3 Performances comparées des méthodes

Nous comparons ici les résultats de la GWR seule et ceux de la GWR augmentée par SOM. Nous suivons le protocole général présenté au chapitre 3 : les prix sont prédits chaque mois en prenant en compte 4 ans de données antérieures, et les erreurs relatives sur les transactions observées chaque mois sont calculées. On différencie la période T_{opt}=[2016-01, 2018-12] où les paramètres σ et γ sont calibrés et la période T_{val}=[2019-01, 2021-01]. Les volumes cumulés des ensembles de transactions tests sur la période T_{opt} sont de 64216 et 581 à Paris et aux Lilas respectivement, ceux sur la période T_{val} sont de 40075 et 392 à Paris et aux Lilas respectivement. On affiche les erreurs de prédiction pour Paris et Les Lilas sur la figure 4.11.



(a) Boîtes à moustache des erreurs de prédiction pour la ville des Lilas



(b) Boîtes à moustache des erreurs de prédiction pour la ville de Paris

FIGURE 4.11 – Distributions des erreurs de prédiction aux Lilas et à Paris : en bleu pour la période $T_{\text{opt}} = [2016-01, 2018-12]$, en vert pour la période $T_{\text{val}} = [2019-01, 2021-01]$

L'information apportée par l'algorithme SOM telle que nous l'avons utilisée ne semble pas améliorer significativement les performances — mais une des limitations ici est que les données INSEE employées n'étaient sans doute pas assez récentes.

4.3 Discussion

L'utilisation de SOM permet de recueillir des informations à partir d'un vaste corpus de données socio-économiques afin d'en faire ressortir la structure socio-spatiale qui est en lien avec la dynamique des prix de l'immobilier. Notre méthode reflète des informations difficilement accessibles par d'autres moyens, surtout si l'on n'a pas, ou si l'on ne peut pas, avoir une connaissance intime de la ville considérée : le type et la qualité des bâtiments, l'ambiance d'un quartier, s'il sera bientôt très recherché ou non, etc. De grandes quantités de variables socio-économiques fonctionnent comme proxy pour ces informations, à condition de pouvoir les exploiter à l'aide de méthodes d'apprentissage automatique.

La combinaison des distances sur la carte SOM avec les distances géographiques permet d'obtenir des cartes de prix qui correspondent plus à une réalité de diffusion de prix qu'avec le modèle de base (du moins sur les marchés immobiliers où nous avons testé notre méthode). Néanmoins, les performances du modèle mesurées par les erreurs de prédiction ne reflètent pas l'apport de SOM dans la GWR. Les performances sont similaires sur les villes testées.

Parmi les questions ouvertes figurent celle de la définition des poids qui sont issus du croisement entre les deux types de distances utilisées, et celle de l'interprétabilité des coefficients de régression ainsi obtenus. En effet, on fait le choix d'appliquer une fonction gaussienne pour transformer les distances géographiques et SOM en poids car c'est la fonction la plus utilisée dans la littérature. La recherche des paramètres de largeurs des noyaux étant faite par *grid search*, on pourrait en faire de même pour le choix de la fonction de poids. Aussi, le choix naturel a été de multiplier les poids géographiques et les poids SOM mais il est possible d'imaginer d'autres formules. Par exemple faire en sorte qu'un poids géographique faible puisse être corrigé un poids SOM fort. Tel que défini aujourd'hui, un point éloigné géographiquement ne pourra jamais avoir un poids final aussi fort qu'un point proche géographiquement, même avec un poids SOM égal à 1. Du point de vue de l'interprétabilité, si on étudie l'effet du nombre de pièces sur le prix et si on définit comme bien standard un appartement de deux pièces, alors le coefficient lié à la variable « une pièce » s'interprète comme l'effet marginal d'avoir une pièce plutôt que deux. Qu'en est-il des nouveaux coefficients ?

L'application sur les deux villes d'Ile-de-France, Paris et Les Lilas, montre l'intérêt d'utiliser SOM pour la détection de frontières socio-géographiques mais il serait intéressant de voir l'apport de SOM dans un contexte de données peu denses. En effet, la classification pourrait être utile dans le cas où l'on veut estimer une localisation où il n'y a aucun point observé dans le voisinage immédiat et où les observations les plus proches appartiennent à des clusters très différents du point de vue de SOM. Dans un contexte d'information rare qu'est celui du marché de l'immobilier, l'impact de SOM pourrait être important. Nous montrerons d'ailleurs au chapitre 7 comment cette idée a conduit à la mise en production de nouveaux indices désormais disponibles sur le site de MeilleursAgents. Mais avant ceci, nous examinons dans le chapitre suivant une variante des techniques de krigeage, et dans le chapitre 6 une toute nouvelle méthode de prédiction des prix.

Chapitre 5

Krigeage x SOM

Si la localisation d'un logement a un effet sur son prix, alors elle impactera aussi le prix des logements voisins. Il est alors intéressant d'étudier l'autocorrélation spatiale des observations. Les outils issus de la géostatistique permettent d'étudier des phénomènes spatiaux à travers la construction et l'analyse de variogrammes [25]. Les réalisations des phénomènes spatiaux étant uniques, il est alors nécessaire d'utiliser la modélisation. La valeur des estimateurs dépend des observations et de la structure d'autocorrélation spatiale, donnée par le variogramme. On présente ici le krigeage qui est le prédicteur linéaire qui garantit une variance minimum.

De la même manière que la GWR seule, le krigeage valorise deux appartements proches géographiquement seulement ces deux mêmes appartements peuvent se situer de part et d'autre d'une rue où la sociologie sera très différente. Ainsi, considérer uniquement la distance géographique et la corrélation spatiale entre les observations biaiserait l'estimation. Comme au chapitre précédent, on cherche donc à améliorer la méthode en utilisant la proximité socio-économique en plus de la proximité spatiale. On cherche ici à corriger les poids lors de l'interpolation de manière à ce qu'ils ne contiennent plus seulement l'information géographique, mais aussi une notion de proximité socio-économique.

Ce chapitre présente dans un premier temps les semivariogrammes et les équations de krigeage pour le cas particulier d'une application pour l'estimation immobilière. Nous décrivons ensuite notre nouvelle méthode puis les modèles sont appliqués sur les mêmes villes qu'au chapitre précédent (Paris et Les Lilas).

5.1 Processus d'interpolation spatiale : le krigeage

La géostatistique est l'étude de variables réparties dans l'espace dans le but d'analyser les relations qui peuvent exister entre elles. Elle est d'abord utilisée dans le traitement des gisements miniers par l'ingénieur Danie G. Krige puis s'est développée grace aux travaux de George Matheron en 1965 [25]. Elle s'est par la suite étendue à d'autres domaines tels que la biologie [42].

On applique ici l'analyse géostatistique dans le domaine de l'estimation des prix de l'immobilier. Comme vu dans le chapitre 2, le processus observé doit répondre à des hypothèses pour pouvoir utiliser un certain nombre d'outils, notamment celui qui nous intéresse ici est le semivariogramme.

On fait l'hypothèse que $P(.)$ est un processus stationnaire intrinsèque (voir définition 1.2.1.3), de variogramme $\gamma(h)$ et de moyenne m inconune. Pour chaque transaction $i, i = 1, ..., N$, on a connaissance de son prix P_i aussi bien que de sa position géographique. Deux transactions i et j peuvent se situer sur la même localisation $u = (x, y)$ ce qui entraine des problèmes lors de l'inversion de matrice pour le calcul des poids de krigeage (équation 1.16). En effet, si $h = 0$ alors $\gamma(h) = 0$ et donc la matrice des semivariances entre les observations n'est pas de rang plein. Pour pallier ce problème, on produit un léger décalage sur les coordonnées spatiales de i et j afin que la distance entre i et j soit différente de zéro, mais reste largement négligeable face à d'autres distances observables. On peut définir les nouvelles localisations pour i et j par $u_i = (x_i, y_i)$ et $u_j = (x_j, y_j)$ respectivement telles que :

$$||u_i - u_j||\delta < ||v - s|| \tag{5.1}$$

Pour v et s deux localisations très proches, δ se compte en centaine. Les nouvelles coordonnées de i sont calculées à partir de :

$$u_i = \begin{cases} x_i = x + \varepsilon_i^x \\ y_i = y + \varepsilon_i^y \end{cases}$$

où ε_i est une réalisation d'un processus gaussien centré $\mathcal{N}(\mu, \sigma)$ avec $\sigma = 0.01$. Le prix de la transaction i peut-être maintenant décrit directement par $P(u_i)$.

On peut donc construire le semivariogramme empirique à partir des données

filtrées, normalisées actualisées comme décrites dans le chapitre 3 et légèrement décalées spatialement (équation 1.11). On ajuste ensuite un variogramme théorique exponentiel (1.12) au semivariogramme empirique. Un exemple de semivariogramme construit à partir des données filtrées, normalisées et actualisées au 1er janvier 2021 aux Lilas est donné par la Figure 5.1.

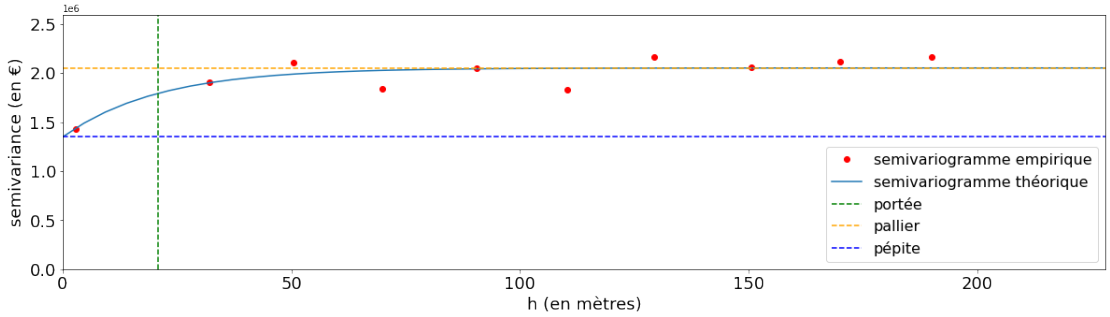


FIGURE 5.1 – Semivariogramme empirique et théorique aux Lilas, 1er janvier 2021. Chaque point du semivariogramme empirique correspond à la moyenne des distances (en abscisses) et à la moyenne des semivariances (en ordonnée) pour chaque pas h appelé lag step. Ici le lag step est égal à 25 mètres et on calcule les semivariances jusqu'à une distance de 200 mètres (cutoff). On obtient une valeur de pépite $c_0 = 1349346$, portée $a = 20$ et palier c_s=2052838.

Le krigeage permet d'avoir des estimations des prix de l'immobilier qui dépendent de la distance géographique, mais aussi de la corrélation spatiale des observations entres elles. Cependant cela reste un modèle de diffusion spatiale où les seules informations à disposition sont celles du prix de la transaction et de sa localisation. Pour pallier cette limite, nous décidons d'utiliser d'autres variables qui décriraient le quartier. Exactement de la même manière que pour le chapitre précédent, on applique un clustering basé sur ces variables supplémentaires afin de mieux estimer les appartements de la zone étudiée.

5.2 Krigeage x SOM

On désigne le nombre de blocs dans une ville par n et $y_j \in \mathbb{R}^q$ le vecteur de variables socio-économiques du bloc j pour $j = 1, ..., n$. On applique l'algorithme SOM sur tous les y et on obtient les clusters $C_1, ..., C_K$. Pour chaque transaction i, on cherche le bloc j dont elle fait partit, désigné par $j(i)$ et le vecteur y associé à i est noté $y_{j(i)}$ avec $y \in C_k$. Par simplicité d'écriture, on désigne par P_i^k le prix de la

transaction i appartenant à la classe C_k.

Dans chaque classe, il existe une moyenne m_k et une variance σ_k^2 avec :

$$m_k = \frac{1}{N_k} \sum_{i \in C_k} P_i^k \qquad \sigma_k^2 = \frac{1}{N_k} \sum_{i \in C_k} \left(P_i^k - m_k\right)^2 = \frac{1}{N_k} \sum_{i \in C_k} \left(P_i^k\right)^2 - (m_k)^2$$

On note d_{kl} la distance entre la classe k et la classe l dans l'espace SOM, avec $1 \leq k, l \leq K$. On aura $d_{kk} = 0$ et $d_{kl} = d_{lk}$.

5.2.1 Variogramme empirique et covariance

On veut construire un variogramme et une fonction de covariance ayant un sens dans l'espace SOM. Pour cela, calculons le semivariogramme entre deux classes k et l :

$$\begin{aligned}
\Gamma_X(d_{kl}) &= \frac{1}{N_k} \frac{1}{N_l} \sum_{i \in C_k} \sum_{j \in C_l} \frac{1}{2} [P_i^k - P_j^l]^2 \\
&= \frac{1}{N_k} \frac{1}{N_l} \sum_{i \in C_k} \sum_{j \in C_l} \frac{1}{2} [P_i^{k\,2} + P_j^{l\,2} - 2 P_i^k P_j^l] \\
&= \frac{1}{2N_k} \sum_{i \in C_k} (P_i^k)^2 + \frac{1}{2N_l} \sum_{j \in C_l} (P_j^l)^2 - \frac{1}{N_k} \frac{1}{N_l} \sum_{i \in C_k} P_i^k \sum_{j \in C_l} P_j^l \\
&= \frac{1}{2} (\sigma_k^2 + m_k^2 + \sigma_l^2 + m_l^2) - m_k m_l \\
&= \frac{1}{2} (\sigma_k^2 + \sigma_l^2) + \frac{1}{2} (m_k - m_l)^2
\end{aligned}$$

Plusieurs observations peuvent être faites :

1. Il y a deux termes dans (1) : le premier correspond à la moyenne des variances dans les deux classes ; le deuxième correspond aux semi-écarts quadratiques des moyennes dans les deux classes. Le calcul est donc très rapide, car il ne dépend que de deux valeurs par classes.

2. Lorsque $k = l$, on obtient ;

$$\Gamma_X(d_{kk}) = \sigma_k^2$$

Comme $d_{kk} = 0$ par construction, il vient que le variogramme pour $h = 0$ est pépitique, et non stationnaire, puisque la valeur dépend de la classe k.

3. Afin de rendre le variogramme stationnaire, on normalise les valeurs P_i^k par σ_k et on pose : $Y_i^k = \frac{P_i^k}{\sigma_k}$. Alors, il est aisé de montrer que :

$$\Gamma_Y(d_{kl}) = 1 + \frac{1}{2}\left(\frac{m_k}{\sigma_k} - \frac{m_l}{\sigma_l}\right)^2$$

On a parfois besoin de valeurs du semivariogramme qui ne sont pas définies dans le jeu données et c'est en partie la raison pour laquelle on ajuste un semivariogramme théorique. Or il y a un nombre fini de classes dans l'espace SOM, on connaît donc de façon complète la matrice de covariance. Il y a néanmoins un cas de figure où ce n'est pas vérifié : celui où il n'y a aucune observation appartenant au cluster k_0 et où on voudrait estimer une valeur de $P(.)$ pour une localisation u_0 appartenant au cluster C_{k_0}. Pour pouvoir avoir la semivariance entre la classe k_0 et n'importe quelle autre classe k il faudrait passer par un variogramme théorique Γ_Y^* et récupérer la valeur $\Gamma_Y^*(d_{k_0k})$. En excluant ce cas de figure, l'information de classe suffit et on peut s'abstraire de la distance SOM pour les calculs qui suivent. On peut alors se demander l'intérêt d'utiliser la classification de Kohonen plutôt qu'une autre classification. Il est double : être exactement dans le même cas de figure que pour le chapitre 4 permet de comparer les nouvelles méthodes entre elles (car on reprend les mêmes classifications dans l'application). Aussi, on verra par la suite que la distance dans l'espace SOM apporte une autre grille de lecture et permet de prendre du recul sur les analyses qui vont suivre.

On définit la covariance normalisée entre la classe k et la classe l par :

$$C_Y(kl) = K - \Gamma_Y(kl) \tag{5.2}$$

où $K = \max_{kl}\Gamma_Y(kl)$.
On obtient la covariance de $C_X^{SOM}(kl)$ en multipliant par les écarts-types des classes

respectives σ_k et σ_l. On a alors :

$$C_X^{SOM}(kl) = \sigma_k \sigma_l C_Y(kl)$$

On affiche le semivariogramme $\Gamma_Y(kl)$ dans l'espace SOM ainsi que la covariance des données stationarisées $C_Y(kl)$ et la covariance $C_X^{SOM}(kl)$ sur les figures 5.2, 5.3 et 5.4.

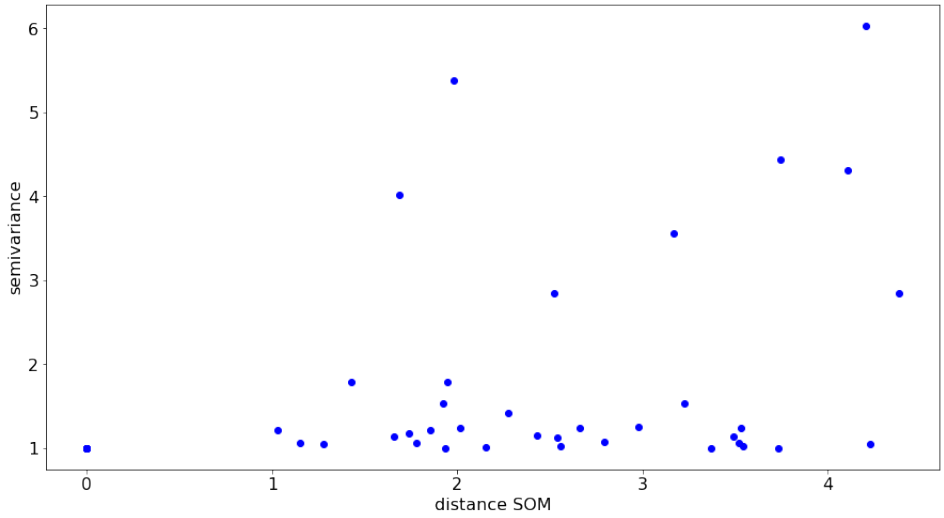


FIGURE 5.2 – Semivariogramme empirique SOM des données stationnarisées aux Lilas, 1er janvier 2021.

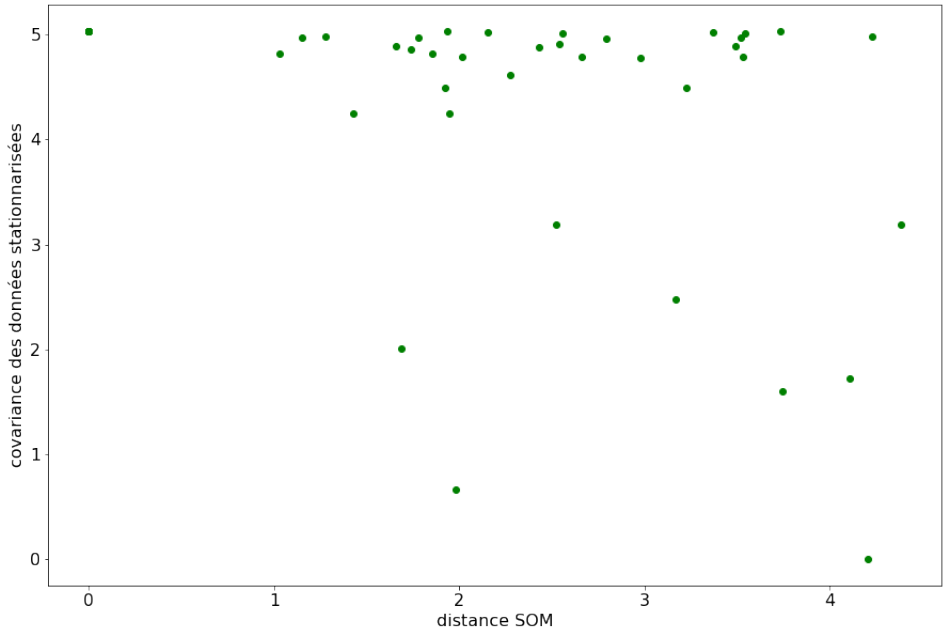


FIGURE 5.3 – Covariances SOM des données stationnarisées SOM aux Lilas, 1er janvier 2021.

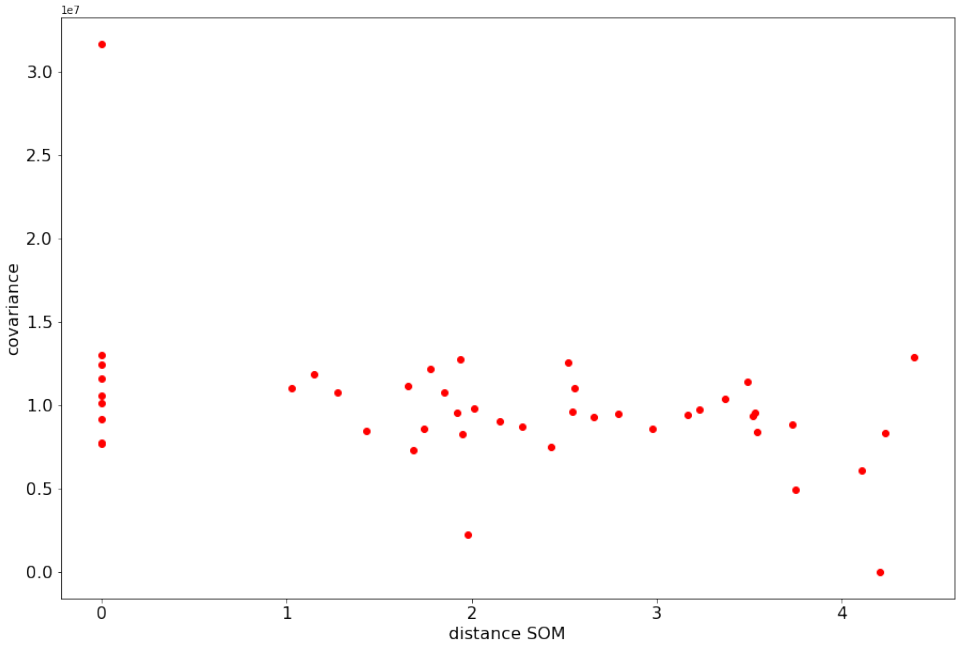


FIGURE 5.4 – Covariances SOM aux Lilas, 1er janvier 2021.

5.2.2 Covariance jointe

Le but de cette section est de croiser l'information apportée par la classification SOM et l'information géographique via les matrices de covariances. Il faut donc déterminer la matrice de covariance spatiale à partir du variogramme.

On utilise la propriété vue dans le cours d'introduction à la géostatistique donné par Denis Allard [1] sur les variogrammes bornés :

Si un champ aléatoire $P(x)$ vérifiant les hypothèses intrinsèques possède un variogramme $\gamma(h)$ borné, c'est-à-dire tel que :

$$\lim_{||h||\to\infty}\gamma(h) = \gamma(\infty) < \infty,$$

alors $P(x)$ est un champ aléatoire stationnaire d'ordre 2.

Le variogramme exponentiel faisant partie de la classe des variogrammes bornés, on peut appliquer cette proposition et définir la covariance spatiale par :

$$C_X^{GEO} = \gamma^{"}(\infty) - \gamma^{"}(h) \tag{5.3}$$

Où $\gamma^{"}(h)$ est le variogramme théorique ajusté au variogramme empirique spatial construit à partir de paires d'observations appartenant au même cluster uniquement. En adaptant l'équation 1.11 il est défini par :

$$\hat{\gamma}^{"}(h) = \frac{1}{2N^{"}(h)} \sum_{\substack{i,j:h-\Delta h\leq|u_i-u_j|\leq h+\Delta h \\ C(u_i)=C(u_j)}}^{N^{"}(h)} (P(u_i) - P(u_j))^2 \tag{5.4}$$

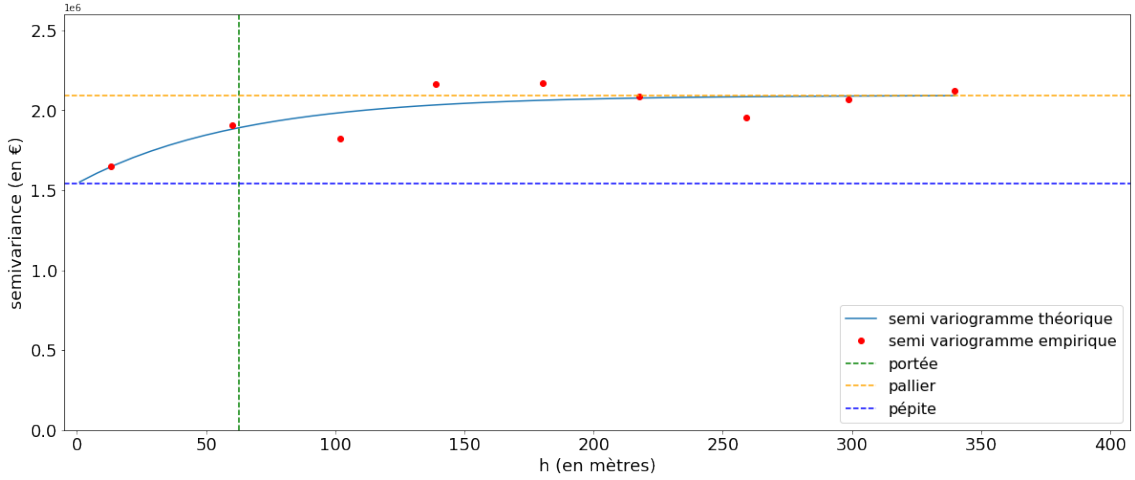
Avec $C(u_i)$ le cluster de la localisation u_i. Il se peut donc qu'il n'y ait pas les mêmes paramètres de portée, pépite et palier que pour un semivariogramme spatial (section 5.1) où l'on ne prend pas en compte la classification issue de SOM. Les valeurs de portée $\hat{a}$, palier $\hat{c}_s$ et pépite $\hat{c}_0$ qui réalisent le minimum de 1.14 sont les paramètres du variogramme théorique $\gamma^{"}(h)$.

La covariance spatiale est donc égale à :

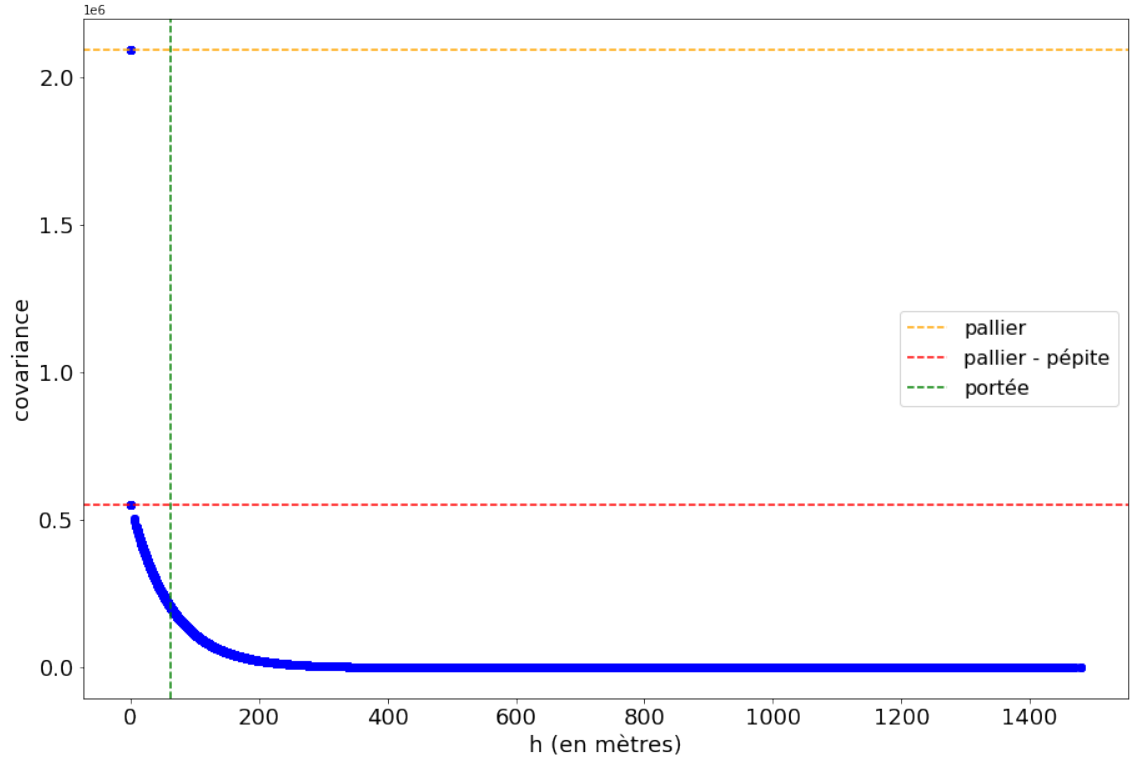
$$C_X^{GEO}(h) = \hat{c}_s - \left(\hat{c}_0 \mathbb{1}_{\{h>0\}} + (\hat{c}_s - \hat{c}_0 \mathbb{1}_{\{h>0\}})(1 - \exp(-h/\hat{a}))\right) \tag{5.5}$$

$$= (\hat{c}_s - \hat{c}_0 \mathbb{1}_{\{h>0\}}) \exp(-h/\hat{a}) \tag{5.6}$$

On affiche le semivariogramme spatial en prenant en compte la classification SOM et la covariance spatiale correspondante aux Lilas dans la Figure 5.5.



(a) Semivariogrammes empirique et théorique en prenant en compte l'information de clustering, aux Lilas au 1er janvier 2021. Le lag step est égal à 40 mètres et le cutoff à 400. 0n obtient une valeur de pépite $\hat{c_0}$= 1542062, portée $\hat{a}$=63 et palier $\hat{c_s}$ = 2092630.



(b) Covariance spatiale aux Lilas, 1er janvier 2021.

FIGURE 5.5 – Semivariogrammes empirique et théorique aux Lilas au 1er janvier 2021.

On peut maintenant construire une covariance jointe, qui associe les distances dans les deux espaces. On fera une hypothèse de séparabilité, à savoir :

$$C_X(h, kl) = C_X^{GEO}(h) C_X^{SOM}(kl) \tag{5.7}$$

On peut distinguer 3 cas de figure :

1. Si on a $P(u_i)$, $P(u_j)$ avec $C(u_i) = k$, $C(u_j) = l$ les classes des observations i et j respectivement et $|u_i - u_j| = h_{ij} > 0$ alors :

$$C_X(h_{ij}, kl) = \Big((\hat{c}_s - \hat{c}_0) \exp(-h_{ij}/\hat{a})\Big)\Big(\sigma_k \sigma_l (K - \Gamma_Y(kl))\Big) \tag{5.8}$$

2. Si on a $P(u_i)$, $P(u_j)$ avec $C(u_i) = C(u_j) = k$ et $|u_i - u_j| = h_{ij} > 0$ alors :

$$C_X(h_{ij}, kk) = \Big((\hat{c}_s - \hat{c}_0) \exp(-h_{ij}/\hat{a})\Big)\Big({\sigma_k}^2 (K - 1)\Big) \tag{5.9}$$

3. Si on a $P(u_i)$ avec $C(u_i) = k$ et $h_{ij} = 0$ alors :

$$C_X(h_{ij}, kk) = \hat{c}_s\Big({\sigma_k}^2 (K - 1)\Big) \tag{5.10}$$

En reprenant l'équation 1.15, l'estimation de $P(.)$ en un point inobservé u_0 est donné par :

$$P^*(u_0) = \sum_{i=1}^{N} \lambda_i^F P(u_i), \tag{5.11}$$

où λ^F est le N-vecteur de poids finaux donnés par la covariance croisée et défini par :

$$\begin{bmatrix} \lambda_1^F \\ \lambda_2^F \\ \vdots \\ \lambda_N^F \\ m \end{bmatrix} = \begin{bmatrix} C_X(h_{11}, C(u_1)C(u_1)) & \cdots & C_X(h_{1N}, C(u_1)C(u_N)) & 1 \\ C_X(h_{21}, C(u_2)C(u_1)) & \cdots & C_X(h_{2N}, C(u_2)C(u_N)) & 1 \\ \vdots & \vdots & \ddots & \vdots \\ C_X(h_{N1}, C(u_N)C(u_1)) & \cdots & C_X(h_{NN}, C(u_N)C(u_N)) & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} C_X(h_{01}, C(u_0)C(u_1)) \\ C_X(h_{02}, C(u_0)C(u_2)) \\ \vdots \\ C_X(h_{0N}, C(u_0)C(u_N)) \\ 1 \end{bmatrix} \tag{5.12}$$

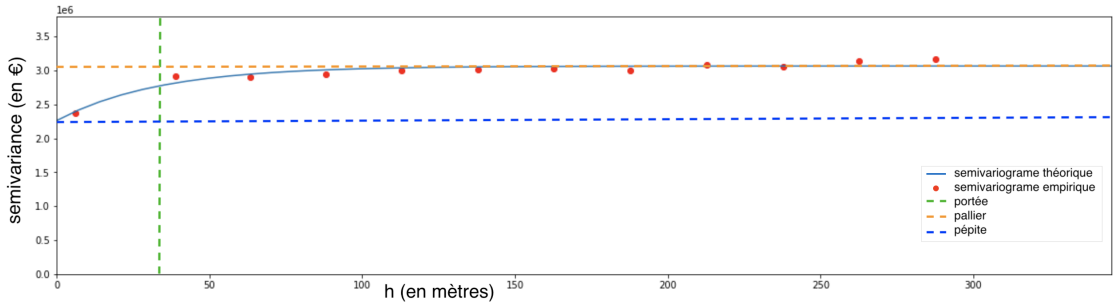
La nouvelle méthode ne permet plus de faire appel aux fonctions du package **gstat** du logiciel **R** pour utiliser le krigeage. Bien qu'il faille inverser la matrice des corrélations entre les observations une seule fois, le processus d'interpolation reste nettement plus long que celui utilisant la fonction **krige**.

5.3 Application sur deux villes réelles

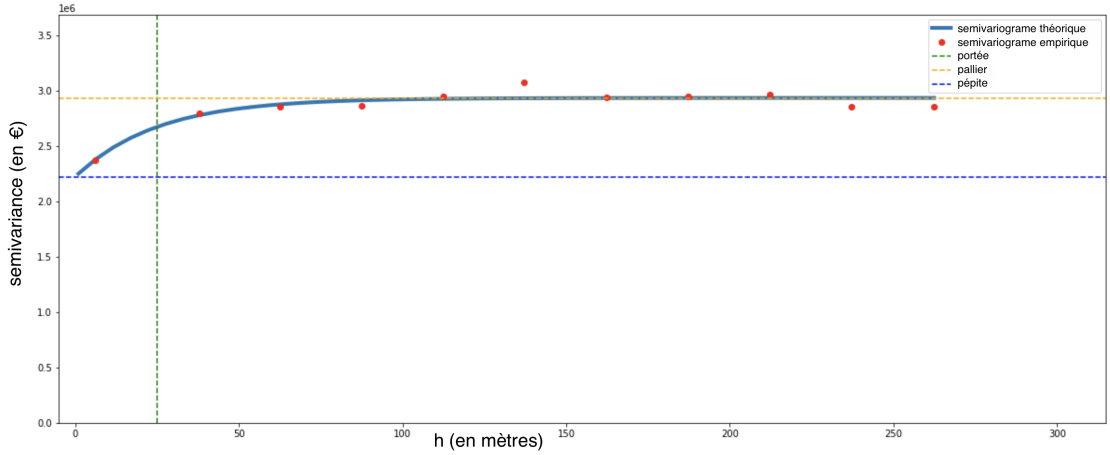
5.3.1 SOM et semivariogrammes

Comme au chapitre 4, on applique cette nouvelle méthode sur Paris et Les Lilas. L'algorithme SOM est appliqué sur les deux villes exactement de la même manière qu'au chapitre précédent. Le même protocole général, présenté au chapitre 3, est utilisé : les prix sont prédits chaque mois en prenant en compte 4 ans de données antérieures, et les erreurs relatives sur les transactions observées chaque mois sont calculés. Il n'y a pas besoin ici d'optimisation sur une période T_{opt} pour fixer des paramètres. Une autre différence est que les données sont non seulement filtrées et actualisées mais aussi ici normalisées à l'aide d'une régression hédonique pour ne travailler que sur le prix au m^2 d'un bien standard, puisque la méthode de krigeage, contrairement à la GWR, ne prend pas elle-même directement en charge les caractéristiques des biens.

On montre sur la figure 5.6 les semivariogrammes empirique et théorique obtenus à Paris pour le cas spatial (équation 1.11) et pour le cas spatial en prenant en compte uniquement les paires d'observations appartenant au même cluster (équation 5.4) :



(a) Semivariogrammes empirique et théorique à Paris au 1er janvier 2021. Les paramètres sont les suivants : portée = 33, palier = 3064672, pépite = 2261080.



(b) Semivariogrammes empirique et théorique à Paris au 1er janvier 2021 pour un échantillon aléatoire de 30000 observations. Les calculs sont ensuite effectués en prenant en compte l'information de classe. Les paramètres sont les suivants : portée = 25, palier = 2935140, pépite = 2222555.

FIGURE 5.6 – Semivariogrammes empiriques et théoriques à Paris au 1er janvier 2021.

Les paramètres des deux semivariogrammes ne sont pas exactement les mêmes, mais sont très semblables (33 mètres vs 25 mètres pour la portée). Cela du fait que dans le deuxième, le calcul du semivariogramme empirique a dû être fait sur un sous-échantillon de D_{train} car le volume de données (Card(D_{train})=81943) est trop important. De plus, on ne retient que les paires d'observations appartenant au même cluster à partir de ce sous-échantillon (pour le cas de Paris uniquement).

Des techniques de calcul de distances peu coûteux sont aujourd'hui mis en place chez MeilleursAgents via l'utilisation de **KDTree** et de matrices creuses. Seulement nous n'avons pas pu adapter le code de calcul de distance pour y intégrer la notion de cluster. C'est pour cela que pour pallier les problèmes de mémoire il a fallu faire le

choix de construire le semivariogramme empirique sur un sous-échantillon aléatoire.

Le semivariogramme et la covariance dans l'espace SOM sont calculés chaque mois. La figure 5.7 représente le semivariogramme SOM au 1er janvier 2021.

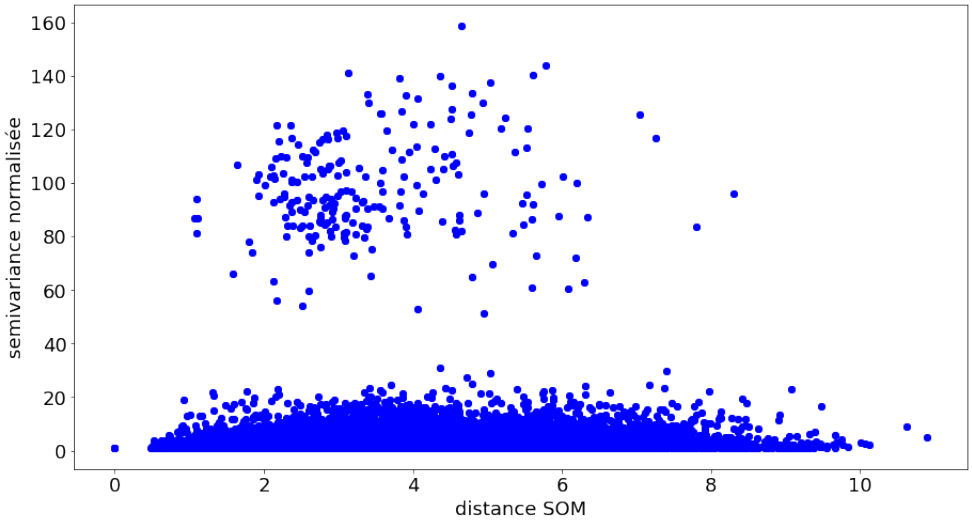


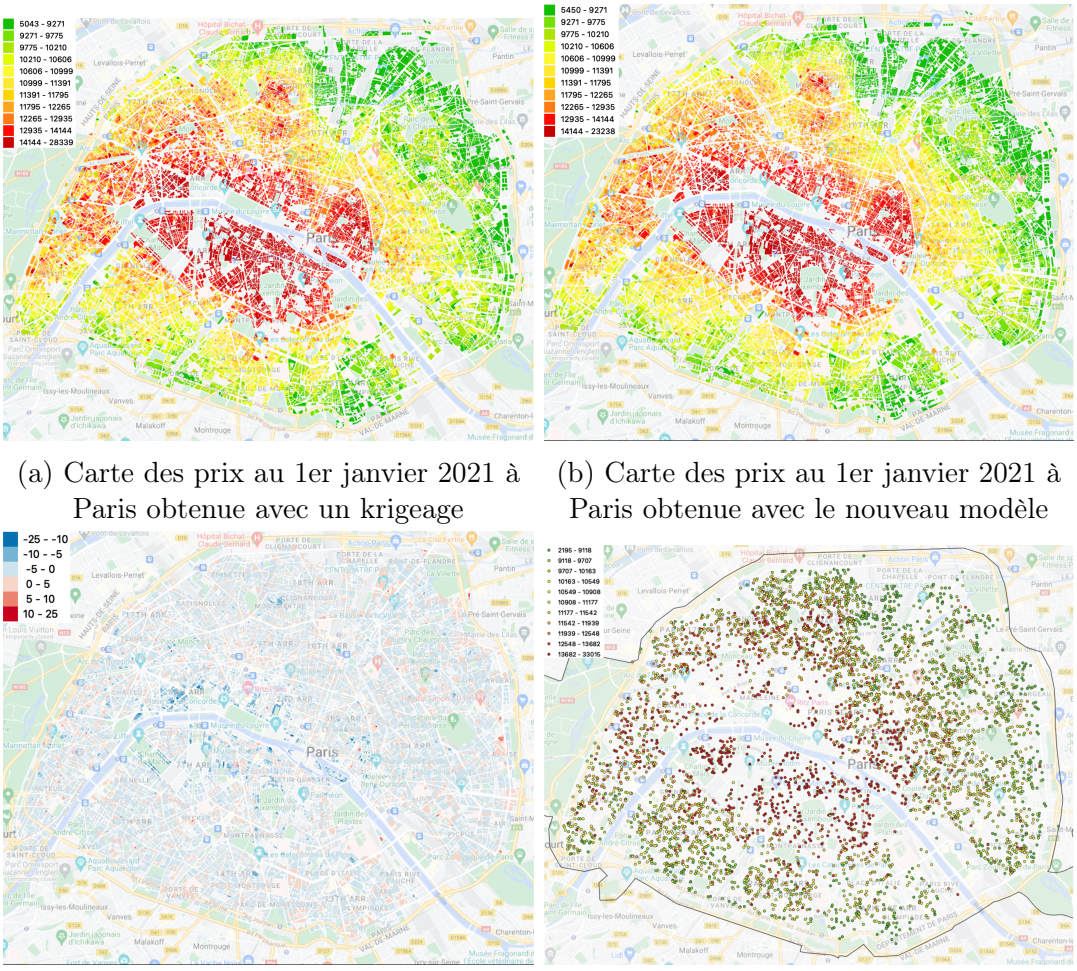
FIGURE 5.7 – Semivariogramme empirique normalisé SOM à Paris au 1er janvier 2021.

Le semivariogramme dans l'espace SOM de Paris (Figure 5.7) et celui des Lilas (Figure 5.2) sont très différents. En effet, celui des Lilas est globalement croissant et on pourrait le modéliser à partir d'un modèle puissance (1.13) : c'est à dire une semivariance faible pour des observations appartenant à des clusters proches (au sens de la distance SOM) et une semivariance plus élevée de manière continue au fur et à mesure que la distance SOM augmente. Il n'y a pas d'effet plateau comme pour le cas spatial, car ce sont les informations liées à la classification qui sont montrées ici. Le semivariogramme de Paris (Figure 5.7), lui, présente deux parties distinctes : les paires d'observations dont la semivariance normalisée est inférieure à 40 et celles pour lesquelles elle est supérieure. Indifféremment de la distance SOM, les prix au sein de deux clusters peuvent être très homogènes (semivariance faible) ou au contraire très différent (semivariance forte). Cela peut soit traduire une mauvaise classification, soit remettre en question l'hypothèse selon laquelle des localisations proches d'un point de vue socio-économique ont des dynamiques de prix de l'immobilier similaires, soit enfin le nombre de clusters est trop faible et donc des localisations présentant des caractéristiques différentes sont regroupées dans des clusters proches sur la grille

SOM. Bien que la notion de distance ne soit pas utilisée dans la suite de nos calculs, le fait d'avoir à disposition la distance SOM permet d'avoir un regard critique sur la classification obtenue et le semivariogramme SOM peut-être utilisé en tant qu'outil d'analyse à part entière pour la classification.

Les prix sont prédits au niveau de la parcelle en prenant en compte son centroïde pour le calcul des distances. Les cartes des prix obtenus pour les modèles de krigeage et krigeagexSOM sur les deux villes de Paris et Les Lilas sont affichés dans les Figures 5.8 et 5.10. On affiche aussi les cartes des différences de prix entre le modèle de base (krigeage) et le nouveau modèle. La différence de prix pour une parcelle i est défini par :

$$\Delta_i(t) = \frac{|P_i(t)^{\text{KrigeagexSOM}} - P_i(t)^{\text{Krigeage}}|}{P_i(t)^{\text{Krigeage}}} \times 100 \tag{5.13}$$



(a) Carte des prix au 1er janvier 2021 à Paris obtenue avec un krigeage

(b) Carte des prix au 1er janvier 2021 à Paris obtenue avec le nouveau modèle

(c) Différence de prix en pourcentage entre le nouveau modèle et le krigeage

(d) Prix au m^2 normalisé des observations du premier trimestre 2021

FIGURE 5.8 – Cartes des prix à Paris aux 1er janvier 2021 et observations au premier trimestre 2021.

Pour Paris, la structure globale de la ville reste la même (Figure 5.8 (a) et (b)) et les prix varient globalement peu d'un modèle à l'autre à part pour certains cas extrêmes où la variation atteint 25% (Figure 5.8 (c)). Pour la ville de Paris et Les Lilas, l'information apportée par l'algorithme SOM telle que nous l'avons utilisée ne semble pas modifier les poids de krigeage de manière significative (Figure 5.9).

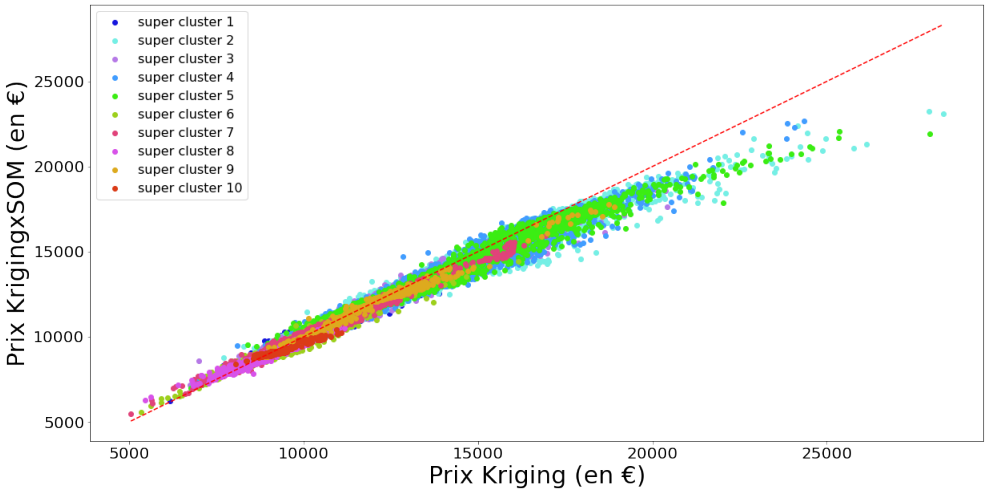
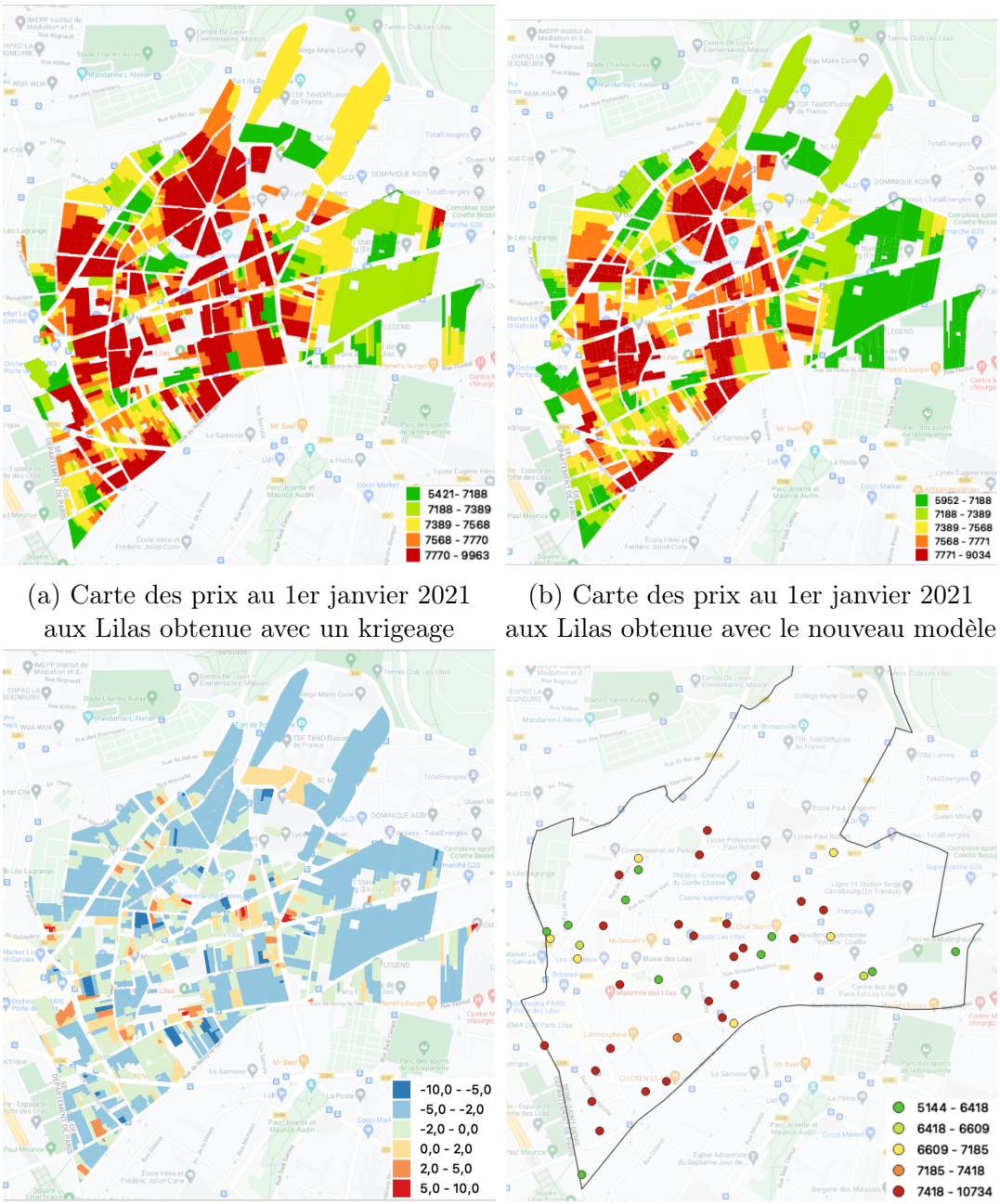


FIGURE 5.9 – Scatter plot des prix à Paris au 1er janvier 2021. En abscisse les prix estimés avec le krigeage simple et en ordonné ceux estimés avec le nouveau modèle.



(a) Carte des prix au 1er janvier 2021 aux Lilas obtenue avec un krigeage

(b) Carte des prix au 1er janvier 2021 aux Lilas obtenue avec le nouveau modèle

(c) Différence de prix en pourcentage entre le nouveau modèle et le krigeage

(d) Différence de prix en pourcentage entre le nouveau modèle et le krigeage

FIGURE 5.10 – Cartes des prix aux Lilas aux 1er janvier 2021 et observations au premier trimestre 2021.

Sur la ville des Lilas, le constat est le même qu'à Paris : la structure spatiale globale reste la même 5.10 (a) et (b)), et les prix varient très peu (5.10(c)). Ici encore, l'effet de l'intégration de la classification dans l'estimation des prix est très

faible (Figure 5.11).

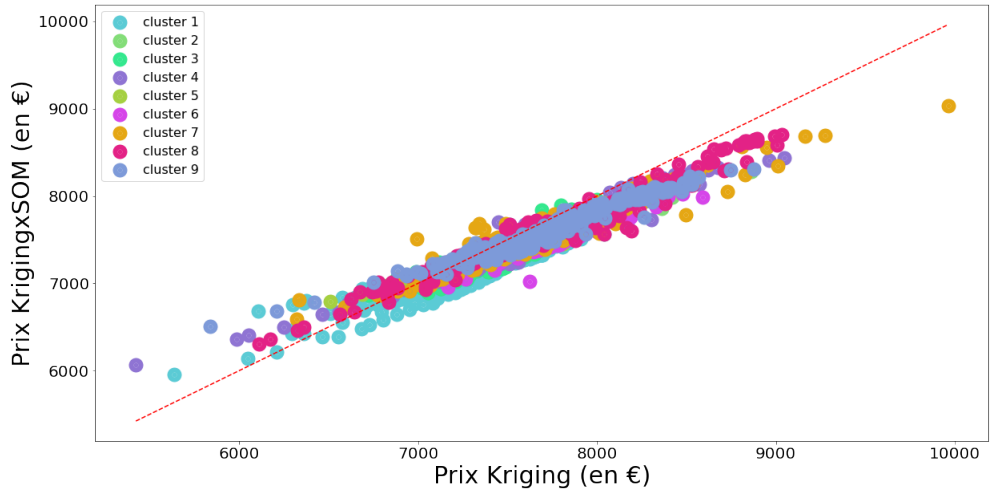
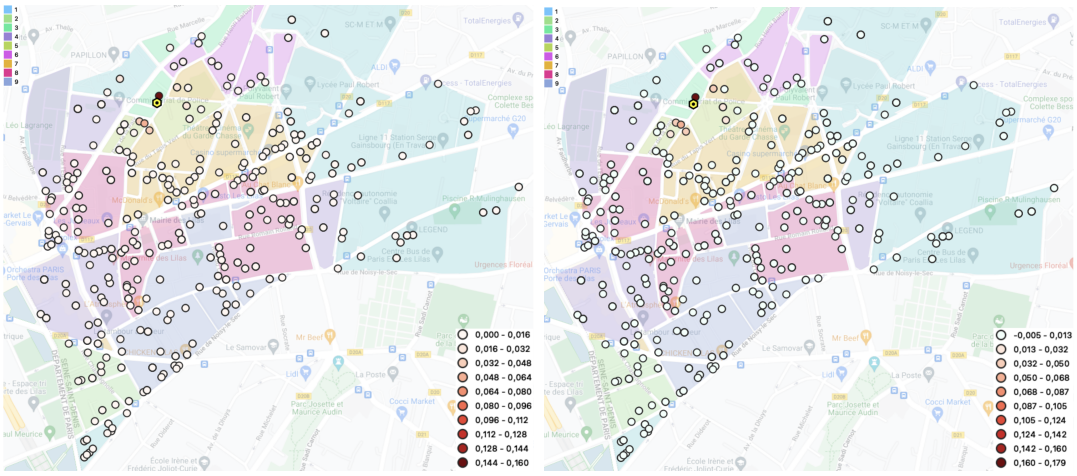


FIGURE 5.11 – Scatter plot des prix aux Lilas au 1er janvier 2021. En abscisse les prix estimés avec le krigeage simple et en ordonné ceux estimés avec le nouveau modèle.

On reprend l'exemple donné dans le chapitre 3 sur l'impact du nouveau modèle sur les poids pour une localisation donnée aux Lilas (Figure 4.5). En reprenant la même parcelle et en appliquant un krigeage simple et celui croisé avec SOM, on obtient les poids affichés dans la Figure 5.12.

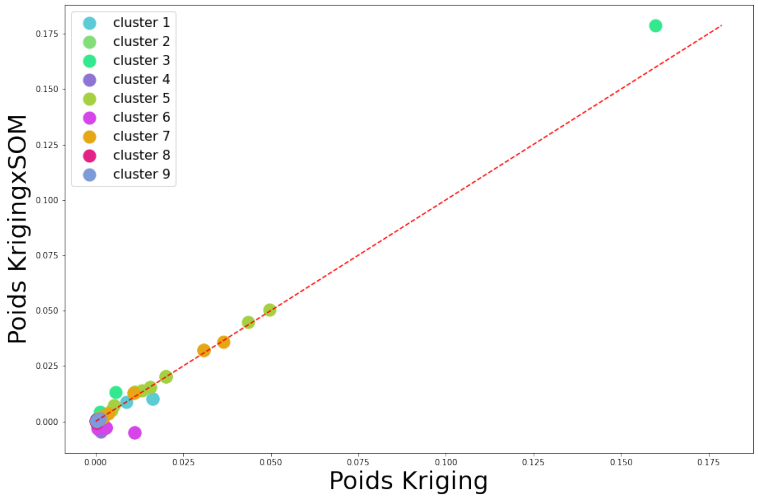


(a) Poids de krigeage pour la parcelle courante (hexagone jaune)

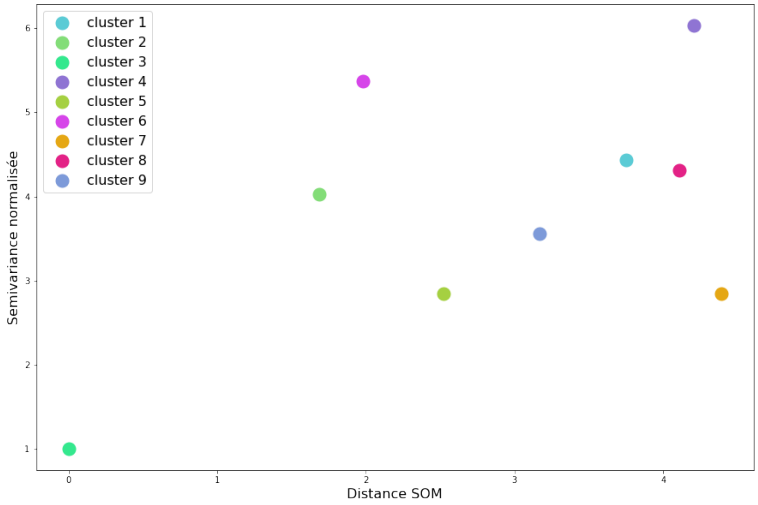
(b) Poids de krigeagexSOM pour la parcelle courante

FIGURE 5.12 – Effet de l'intégration du clustering SOM sur les poids de krigeage aux Lilas au 1er janvier 2021.

L'analyse sur la différence des poids étant difficile à l'oeil nu sur les cartes géographiques (Figure 5.12) on les présente sous forme de nuage de point avec la couleur du cluster pour lequel chaque poids appartient. On étudie ici les poids relatifs à une parcelle appartenant au cluster 3, on affiche donc aussi le semivariogramme entre le cluster 3 et tous les autres clusters.



(a) Scatter plot des poids pour la parcelle courante. En abscisse : les poids formés à partir d'une covariance spatiale, en ordonnée : les poids formés à partir d'une covariance spatiale croisée avec une covariance SOM.



(b) Semivariogramme normalisé pour le cluster 3.

FIGURE 5.13 – Scatter plot des poids et semivariogramme tronqués pour le cluster 3

Les poids relatifs au cluster 6 diminuent avec l'apport de SOM (Figure 5.13(a))

et deviennent relativement plus faible que ceux appartenant au cluster 5 alors que le contraire s'était produit pour la méthode GWRxSOM (Figure 4.6). En effet, si on regarde le semivariogramme tronqué de la Figure 5.13(b), on voit que si le cluster 3 est plus proche du cluster 6 que du cluster 5 au sens de Kohonen, la semivariance entre le cluster 3 et 6 est plus élevée que celle entre le cluster 3 et 5. Le krigeage apporte une dimension en plus par rapport à la GWR qui est la corrélation spatiale des observations entre elles.

5.3.2 Performances comparées des méthodes

Comme au chapitre précédent, nous comparons ici les résultats du krigeage seul et ceux du krigeage augmenté par SOM. Nous suivons toujours le protocole général présenté au chapitre 3 : les prix sont prédits chaque mois en prenant en compte 4 ans de données antérieures, et les erreurs relatives sur les transactions observées chaque mois sont calculées. On pourrait ici ne pas distinguer les deux périodes T_{opt} et T_{val}, car aucun entrainement des paramètres n'est fait, mais on garde cette convention par souci de comparaison avec le chapitre précédent. Les volumes cumulés des ensembles de transactions tests sur la période T_{opt} sont toujours de 64216 et 581 à Paris et aux Lilas respectivement, et ceux sur la période T_{val} de 40075 et 392 à Paris et aux Lilas respectivement. On affiche les erreurs de prédiction pour Paris et Les Lilas sur les figures 5.14 et 5.15

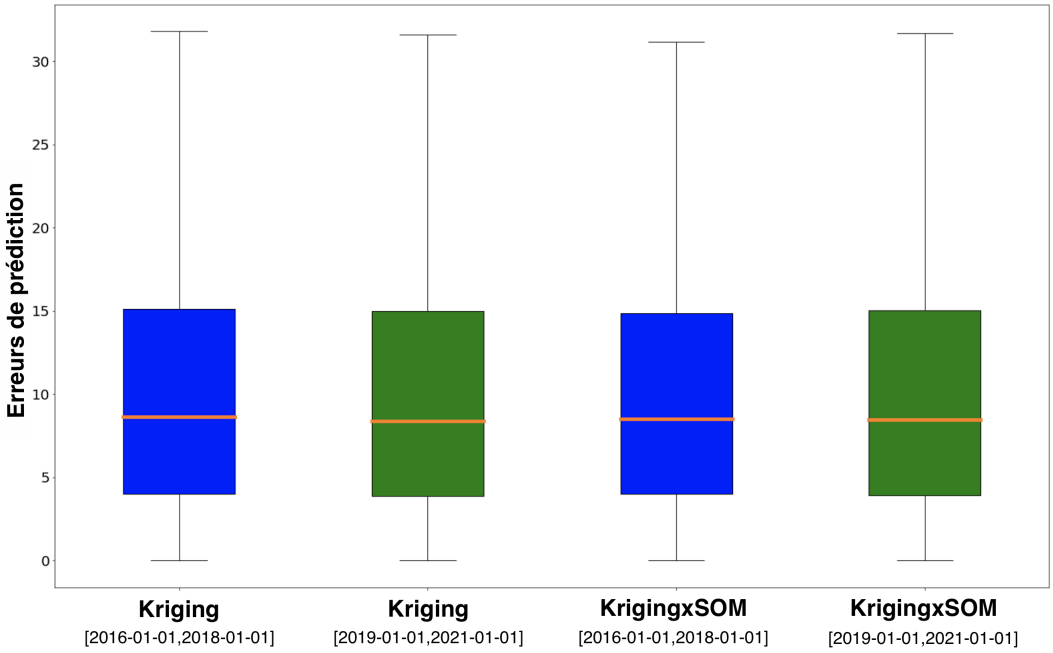


FIGURE 5.14 – Distribution des erreurs de prédiction à Paris

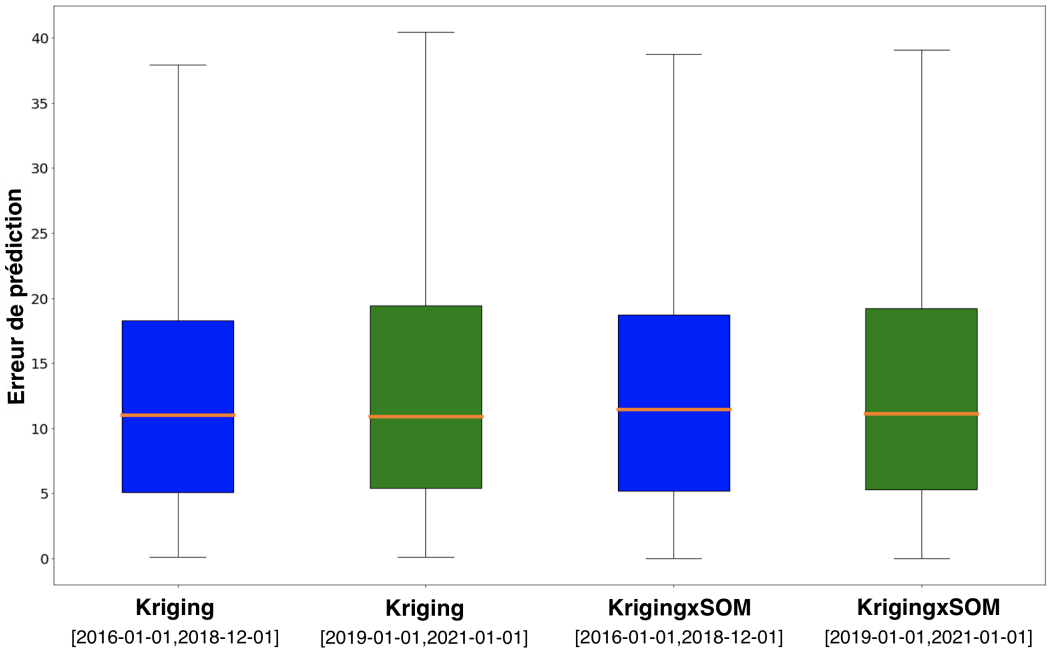


FIGURE 5.15 – Distribution des erreurs de prédiction aux Lilas

Les erreurs de prédiction sur les deux villes d'études sont exactement semblables à celle du modèle de base et donc l'information ajoutée par SOM n'a pas d'impact

du point de vue des performances sur l'échantillon de villes testés — mais, ici aussi, le fait que les données INSEE n'aient pas été suffisamment récentes a pu jouer.

5.4 Discussion

La méthode exposée ici est facile à mettre en place, seulement son exécution en python reste très longue pour Paris (7h). De plus, les résultats sont similaires, voire quasi identiques, à ceux d'un simple krigeage, tant du point de vue de l'aspect des cartes de prix que des performances de prédiction, pour le cas Paris et Les Lilas.

En effet, à la différence des résultats obtenus sur les cartes de prix pour le modèle GWRxSOM, ici, l'impact de SOM sur l'aspect local des cartes est quasi nul. Une explication possible est que la valeur de la portée, pour les deux cas d'application à Paris et Les Lilas, est très faible. Puisque les entités classifiées ici sont les blocs d'une ville, il faudrait que la portée soit au moins supérieure à la distance moyenne entre les blocs pour, effectivement, voir l'apport potentiel d'une covariance jointe. Les seuls points d'estimation où l'on peut espérer une amélioration sont ceux à la frontière entre deux clusters très différents.

Néanmoins, des pistes d'amélioration sont envisageables notamment sur le clustering en tant que tel. Pour que le variogramme SOM ne soit pas pépitique il faut qu'il y ait de l'hétérogénéité entre les classes, autrement dit, il faut que les écarts $(m_k/\sigma_k) - (m_l/\sigma_l)^2$ soient suffisamment grands. On pourrait donc envisager de changer les variables de clustering en passant par un outil de sélection de variables, où la fonction objective serait l'écart entre classes. Une autre solution pourrait être de changer la taille de la grille SOM, par exemple à Paris, où il y a sans doute des cas où des blocs sont rassemblés au sein de la même classe, ou dans des classes similaires alors qu'ils ne décrivent pas les mêmes dynamiques de prix.

Le problème du volume de données reste bloquant pour la mise en place des perspectives d'amélioration. En effet, il faudrait être capable d'adapter le code aujourd'hui en production chez MA qui utilise la recherche de voisins par **KDTree** et l'optimisation mémoire par matrice creuses pour pouvoir construire un semivariogramme avec le volume de données massif que représente la ville de Paris. Il faudrait

ensuite paralléliser le code pour l'estimation en chaque point de la ville afin qu'il se rapproche du temps de calcul de celui de la fonction **krige** du logiciel **R**.

Retenons néanmoins que la représentation des classes SOM sous forme de semi-variogramme semble être un bon outil pour juger de la qualité de la classification.

Chapitre 6

Villes comme réseaux de neurones

Les limites des méthodes présentées dans les deux chapitres précédents nous invitent à un changement de paradigme. Puisqu'une ville peut en définitive, du point de vue des prix de l'immobilier, être vue comme un réseau de localisations entre lesquelles se diffusent de l'information de prix, il est possible de la modéliser directement comme un réseau de neurones, où chaque neurone est caractérisé par le prix au mètre carré de la localisation. Une transaction active un neurone et modifie son prix et celui de ses voisins par répercussion. Les données passées nous permettent donc d'entrainer le réseau afin qu'il apprenne le marché du logement dans la ville.

Les deux grands enjeux ici sont de faire correspondre la géographie de la ville avec la typologie du réseau, puis de faire correspondre les dynamiques du marché avec les règles du réseau à travers les réglages des fonctions de voisinage et de mise à jour, car ce sont celles-ci qui encodent fondamentalement le marché immobilier de la ville. Ce modèle présente l'avantage d'être mathématiquement simple et rapide d'exécution. Plusieurs paramètres et types d'optimisation de paramètres ont pu être testés. Comme pour les chapitres précédents, l'application de ce nouveau modèle se fait sur les villes de Paris et Les Lilas.

6.1 Typologie du réseau : création du graphe

Le cadastre est l'inventaire descriptif et évaluatif des parcelles de terrain et des immeubles bâtis. Une parcelle cadastrale est une portion de terrain appartenant à un même propriétaire, définie par sa géométrie (généralement un polygone) et un

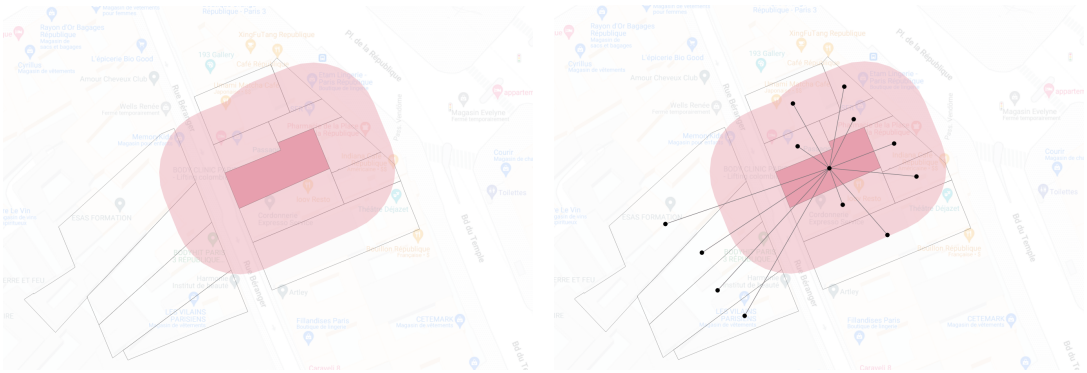
identifiant unique d'après l'APUR (Atelier Parisien d'Urbanisme). Il peut y avoir plusieurs immeubles sur une même parcelle.

Soit i une parcelle, l'ensemble de ses parcelles voisines j du point de vue géographique est noté N_i^{geo} et défini par :

$$N_i^{\text{geo}} = \{j, ||j - i|| \leq r\} \tag{6.1}$$

Avec $r = 25$ mètres la distance maximum pour laquelle des parcelles sont considérées comme voisines. Nous choisissons un rayon de 25 mètres, car c'est, en moyenne, la distance entre deux parcelles adjacentes pour le cas particulier des immeubles résidentiels. On choisit le même rayon pour les deux villes étudiées, à savoir Paris et Les Lilas.

0n associe alors pour chaque parcelle i le noeud correspondant n_i et on définit un graphe $G(M, E)$ où M est l'ensemble des noeuds et E est l'ensemble des arêtes. Si la parcelle $j \in N_i^{\text{geo}}$ alors $\exists e_{ij} = \{i, j\}$ une arête entre les noeuds n_i et n_j. Dans la suite de ce chapitre, on désignera le noeud n_i comme le neurone n_i. Pour schématiser ces propos, on présente la formation des parcelles voisines pour une parcelle courante pour un rayon $r = 25$ mètres dans la Figure 6.1.



(a) Parcelle courante en rose foncé et son buffer de 25 mètres

(b) Définition des parcelles voisines pour la parcelle courante. Les relations de voisinage sont modélisées via les neurones et les arêtes.

FIGURE 6.1 – Exemple de définition du graphe pour une parcelle donnée, à Paris.

Une parcelle est caractérisée par son prix au mètre carré (pm2). Une transaction

active un neurone et modifie son pm2 et celui de ses voisins par répercussion. C'est pourquoi le graphe G est construit de telle manière à conserver la topologie géographique incluant les parcelles résidentielles et non résidentielles. Une parcelle non résidentielle est un bâtiment public (école, hôpital, église ...) ou privé (commerce). La Figure 6.2 représente un zoom du graphe de neurones G à Paris, avec toutes les relations de voisinage.

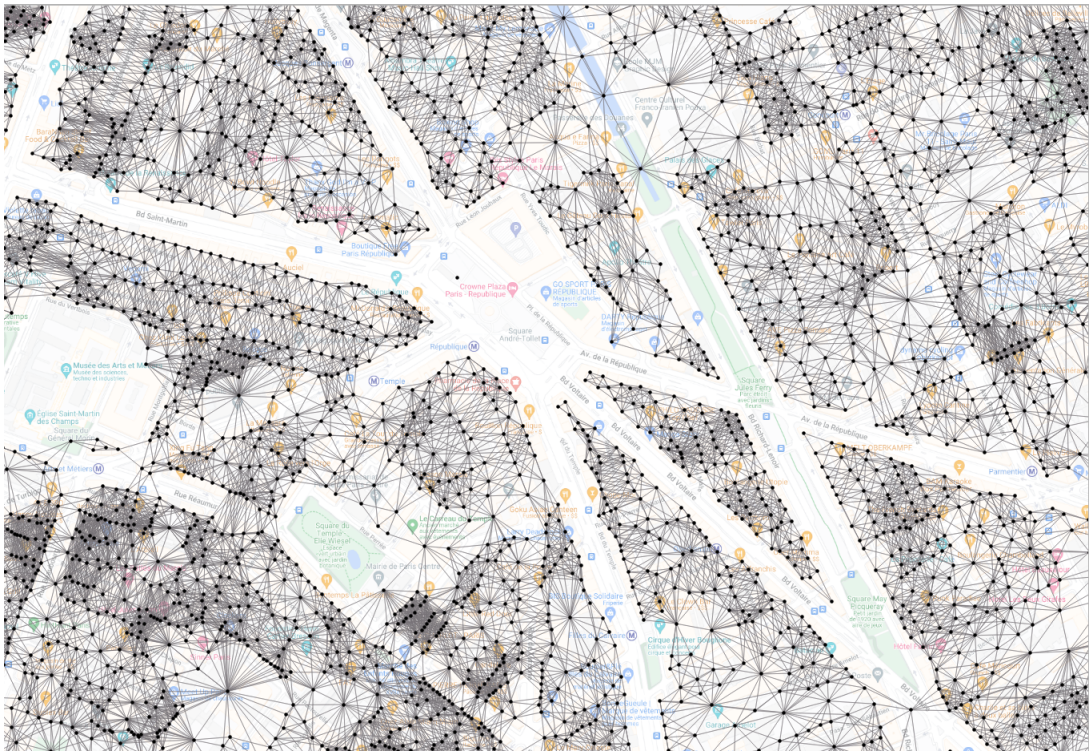


FIGURE 6.2 – Zoom du graphe G pour toutes les parcelles à Paris

Les parcelles définies par la DGFIP ne recouvrent pas tout l'espace apparaissant sur le cadastre (voirie publique, fleuve, boulevard, rue, parc, places ...). On voit ici la limite à ce que le graphe ne soit pas complet : l'information ne peut pas se diffuser de part et d'autre d'un boulevard alors que ces relations de voisinage existent et peuvent être très fortes.

On décide donc de créer des «fausses parcelles» de manière à ce que l'entièreté du territoire soit recouvert. Ces parcelles peuvent être des morceaux de rue, fleuves, parc, etc. D'autre part, la taille des parcelles influe beaucoup sur la détermination du voisinage, c'est pourquoi on choisit de «couper» les parcelles non résidentielles

qui sont trop grandes en taille égale afin de leur donner une taille convenable. Nous choisissons de fixer une taille maximum de 5000m^2, les parcelles dépassant ce seuil seront subdivisées en plusieurs nouvelles parcelles.

On a ainsi le graphe complet final tel que représenté, pour une partie de Paris, sur la Figure 6.3.

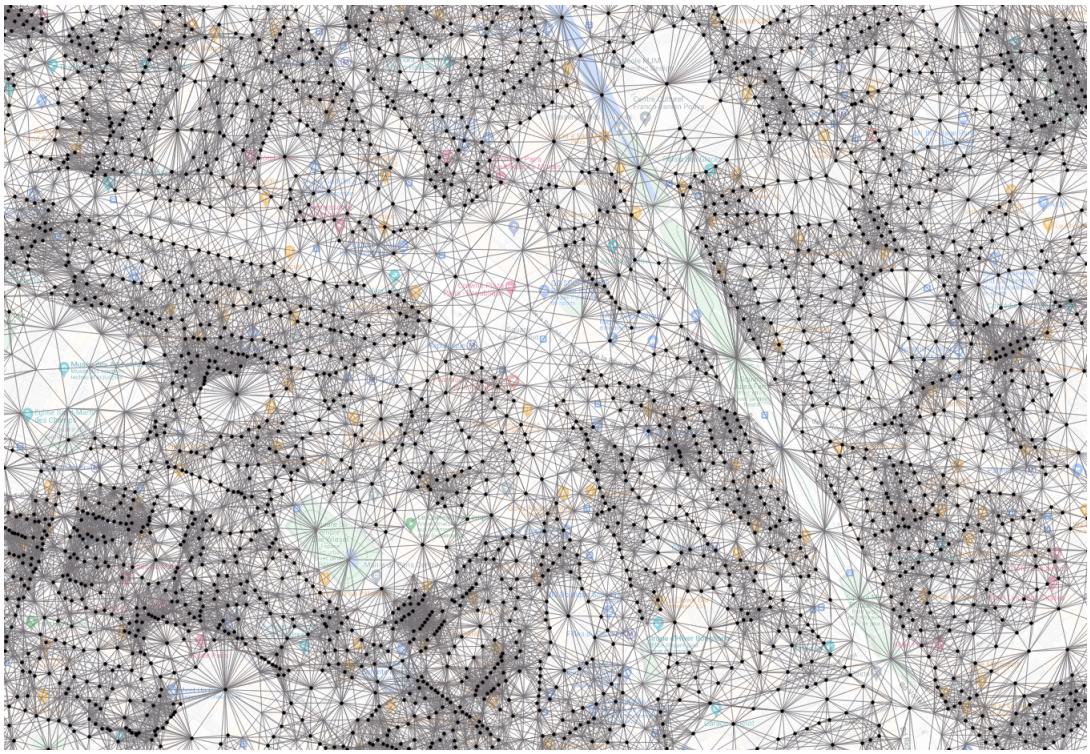


FIGURE 6.3 – Zoom du graphe complet final pour toutes les parcelles à Paris

Le nombre de parcelles voisines dépend de la densité de bâtiments pour une localisation donnée. Par exemple à Paris (Figure 6.4), le nombre de parcelles voisines varie de 1 à 91 pour les zones les plus denses. Pour ces parcelles, cela revient à dire qu'il y a 91 voisins immédiats (c'est-à-dire à distance = 1) sur le graphe.

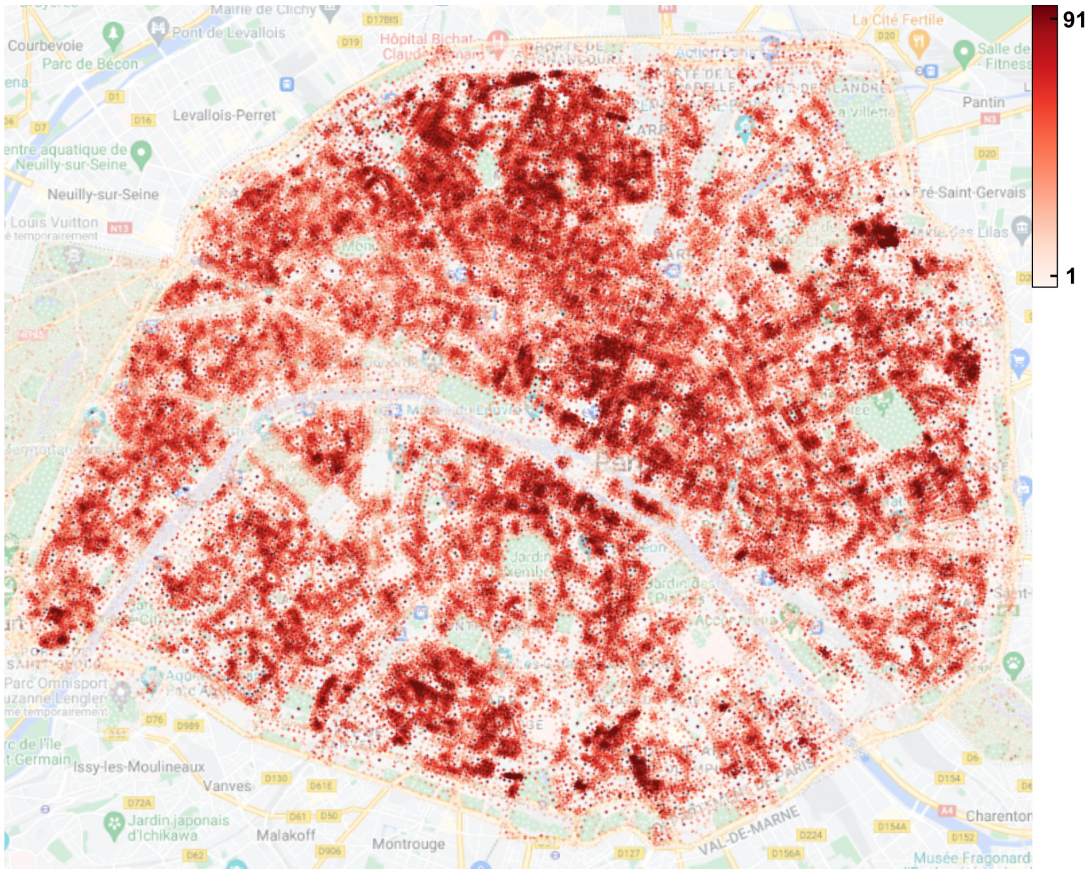


FIGURE 6.4 – Densité du nombre de voisins par parcelle à Paris

Afin d'avoir du recul sur les distances sur le graphe, on représente l'équivalence entre la distance sur le graphe et la distance géographique en mètres (Figure 6.5 exemple à Paris). Le nombre de neurones étant trop élevé pour pouvoir relever les distances en mètres pour chaque paire de neurones de la ville, on tire aléatoirement 100 neurones pour analyser les équivalences de distances. Pour chacun de ces neurones i, on récupère l'ensemble de ses neurones voisins jusqu'à une distance de 60 sur le graphe et on calcule la distance en mètres qui sépare i de chacun de ses voisins. Par exemple, le premier point de la Figure 6.5 représente la distance moyenne en mètre des neurones voisins pour une distance de 1 sur le graphe, pour l'ensemble des 100 neurones pris aléatoirement.

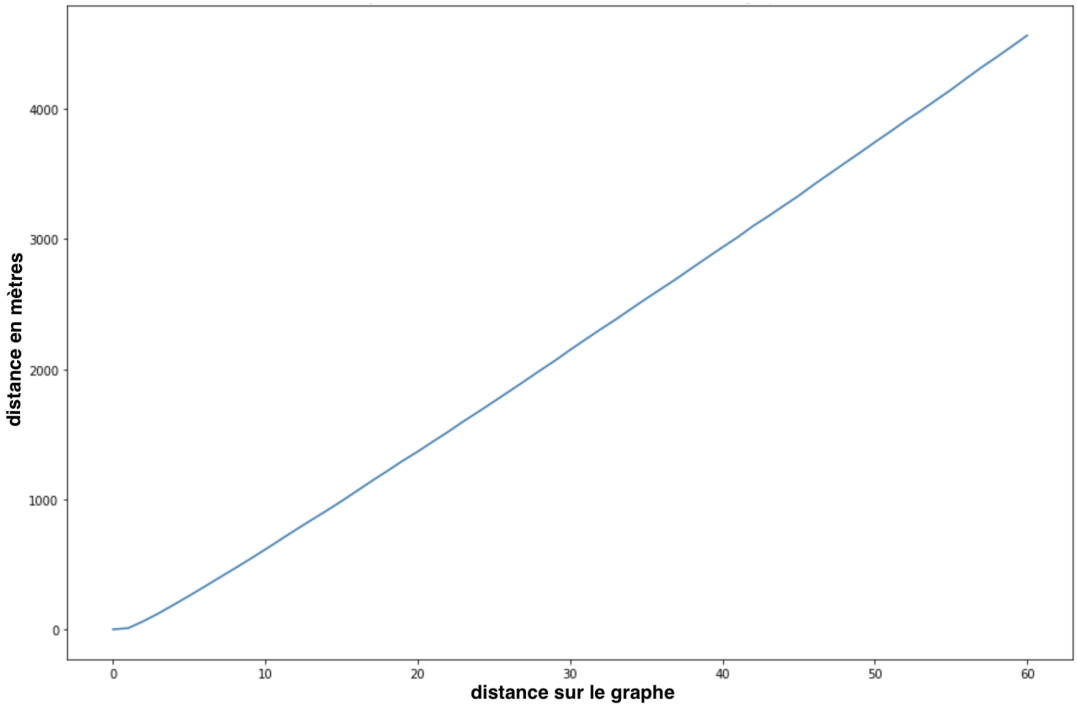


FIGURE 6.5 – Moyenne de la distance en mètre en fonction de la distance sur le graphe pour 100 observations à Paris

6.2 Règles du réseau de neurones

Le graphe représentant la ville va fonctionner simplement comme un réseau de neurones à une couche, dont nous détaillons les règles, usuelles, ci-dessous.

6.2.1 Activation et mise à jour d'un neurone

Au temps t, un certain nombre de transactions s'effectuent dans la ville et leur prix est observé. On note $D(t)$ l'ensemble des transactions qui ont lieu au temps t et $P_a(t)$ le prix observé pour la transaction a, avec $a \in D(t)$. On applique un prétraitement aux transactions : elles sont filtrées et normalisées comme il est décrit dans le chapitre 3.

On désigne par $D_{N_i}(t)$ l'ensemble des transactions dans le voisinage de i au temps t et h_{ai} une fonction de poids décroissante avec la distance (ici la distance sur le graphe). La règle de mise à jour du neurone i au temps $t+1$ est la suivante :

$$P_i(t+1) = \begin{cases} P_i(t) \ + \ \alpha \sum\limits_{a \in D_{N_i}(t)} \dfrac{h_{ai}}{\sum\limits_{b \in D_{N_i}(t)} h_{bi}} \ [P_a(t) - P_i(t)] & \text{si } D_{N_i}(t) \neq \emptyset, \\ P_i(t) & \text{sinon.} \end{cases} \tag{6.2}$$

Si la transaction a se situe sur le neurone i alors $h_{ai} = 0$. Le paramètre $\alpha \in [0, 1]$ est un paramètre contrôlant la sensibilité aux nouvelles observations (semblable au pas d'apprentissage). En effet si $\alpha = 0$ alors $P_i(t+1) = P_i(t)$, l'apport de la nouvelle information au temps t est nul. Si au contraire $\alpha = 1$ alors $P_i(t+1) = P_i(t) + \sum\limits_{a \in T_{N_i}(t)} \frac{h_{ai}}{\sum\limits_{b \in T_{N_i}(t)} h_{bi}} \ [P_a(t) - P_i(t)]$ et donc la totalité de l'information donnée au temps t est intégrée dans le nouveau prix. Les valeurs de α et de n'importe quel paramètre apparaissant dans h_{ai} sont apprises sur les données, par optimisation via un *grid search* comme décrit dans le protocole général au chapitre 3 et appliqué déjà dans le chapitre 4. Différents choix d'optimisation sont examinés dans la section 6.3. La mise à jour du réseau de neurones au temps t est illustrée schématiquement sur la figure 6.6.

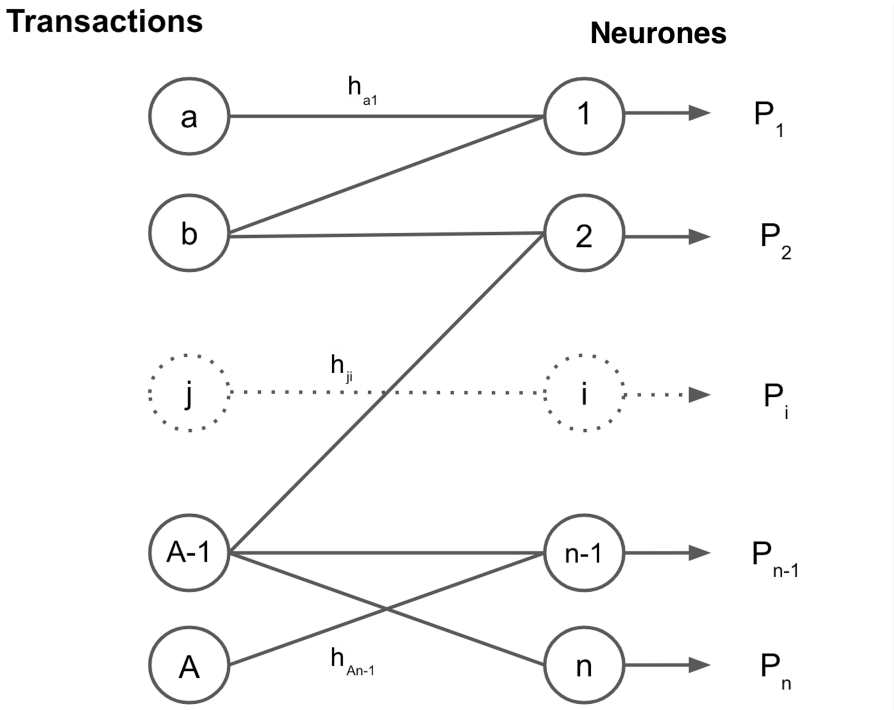


FIGURE 6.6 – Fonctionnement schématique du modèle. Dans cet exemple $P_1(t+1) = P_1(t) + \alpha \left(\frac{h_{a1}[P_a(t) - P_1(t)] + h_{b1}[P_b(t) - P_1(t)]}{h_{a1} + h_{b1}} \right)$

La fonction de voisinage h_{ai} a un rôle central : elle dessine la forme et l'échelle des quartiers dans une ville. On peut pour commencer prendre le voisinage gaussien défini par :

$$h_{ai} = \exp\left(-\frac{d_G(j(a), i)^2}{2\sigma^2} \right) \tag{6.3}$$

où $j(a)$ désigne le neurone j où se situe la transaction a. La distance $d_G(j, i)$ désigne la distance entre les deux neurones j et i sur le graphe G. C'est la longueur d'un plus court chemin entre les neurones j et i, la longueur d'un chemin étant sa longueur en nombre d'arêtes. Les poids issus de h_{ai} dépendent de la largeur du noyau dénoté par σ, que l'on appelle rayon de voisinage sur le graphe G. Plus σ est grand, plus les poids h_{ai} seront grands y compris pour des grandes distances. Cela aura tendance à dessiner des quartiers grossiers. Au contraire, si σ est très petit alors le modèle dessinera des quartiers très fins, prenant seulement en compte l'information ultra-locale. Une autre fonction de voisinage courante est la fonction indicatrice, qui consiste à considérer

un ensemble de neurones autour du neurone i :

$$h_{ai} = \begin{cases} 1 & \text{si } a \in N_i \\ 0 & \text{sinon} \end{cases}, \tag{6.4}$$

où l'ensemble N_i des neurones voisins de i est défini par un rayon de voisinage sur le graphe, rayon que nous pouvons noter également σ par analogie avec la distance caractéristique dans le noyau gaussien. Si N_i est composé des voisins directs de i sur le graphe uniquement (ie $N_i = N_i^{\text{geo}}$) alors $\sigma = 1$. Si on prend en compte les voisins au cinquième degré en prenant le chemin le plus court sur le graphe, alors $\sigma = 5$.

Dans la suite de nos travaux, nous décidons de travailler uniquement avec le voisinage gaussien, par souci de simplicité et après avoir rapidement constaté qu'un voisinage construit par une fonction indicatrice ne donnait pas de grandes différences.

6.2.2 Initialisation du réseau

Il faut assigner une valeur initiale à chaque neurone pour $t = 0$. Pour ce faire, nous considérons 4 ans de données antérieures à la date d'initialisation choisie, en l'occurrence 2016-01. Les observations de 2012-01 à 2015-12 sont normalisées à l'aide d'une régression hédonique comme précédemment, et actualisées à 2016-01 : les prix de transactions antérieures sont multipliés par le coefficient d'évolution entre leur date et 2016-01 [1]. L'ensemble de données ainsi constitué est notre ensemble d'initialisation noté D_{init}, à partir duquel nous allons générer des prix pour tous les neurones du réseau à l'aide d'un noyau gaussien de largeur σ_{geo} :

$$P_i(0) = \sum_{a \in D_{\text{init}}} \frac{\exp\left(-\frac{||r_a - r_i||^2}{2\sigma_{\text{geo}}^2}\right)}{\sum_{b \in D_{\text{init}}} \exp\left(-\frac{||r_b - r_i||^2}{2\sigma_{\text{geo}}^2}\right)} P_a^{actu}(0), \tag{6.5}$$

où r_a et r_i désignent les vecteurs de localisation géographique de la transaction a et du neurone i, respectivement. Nous pouvons choisir σ_{geo} par *grid search* en minimisant l'erreur médiane de prédiction en janvier 2016. Le choix de l'initialisation à

1. Cf chapitre 3 : l'indice des prix de l'immobilier de MeilleursAgents $I(t)$ est utilisé pour calculer un coefficient d'évolution $\delta(t - t_a) = \frac{I(t)}{I(t_a)}$ entre l'instant t_a d'une transaction a et l'instant t où l'on veut l'actualiser.

l'aide d'un noyau gaussien s'est fait du fait de la simplicité de calcul et d'interprétation. Cependant, plusieurs types d'initialisation ont été testés, parmi elles :

- Moyenne : les neurones sont initialisés à partir de la moyenne des prix des observations, les neurones ont donc tous la même valeur sur la totalité du réseau.
- Aléatoire : les neurones sont initialisés en prenant des valeurs aléatoires.
- GWR : les neurones sont initialisés en appliquant une GWR sur les observations.
- Krigeage : les neurones sont initialisés en appliquant un krigeage sur les observations.

L'état initial du réseau pour Paris est représenté sur la figure 6.7, où les prix sont reprojetés sur les parcelles résidentielles.

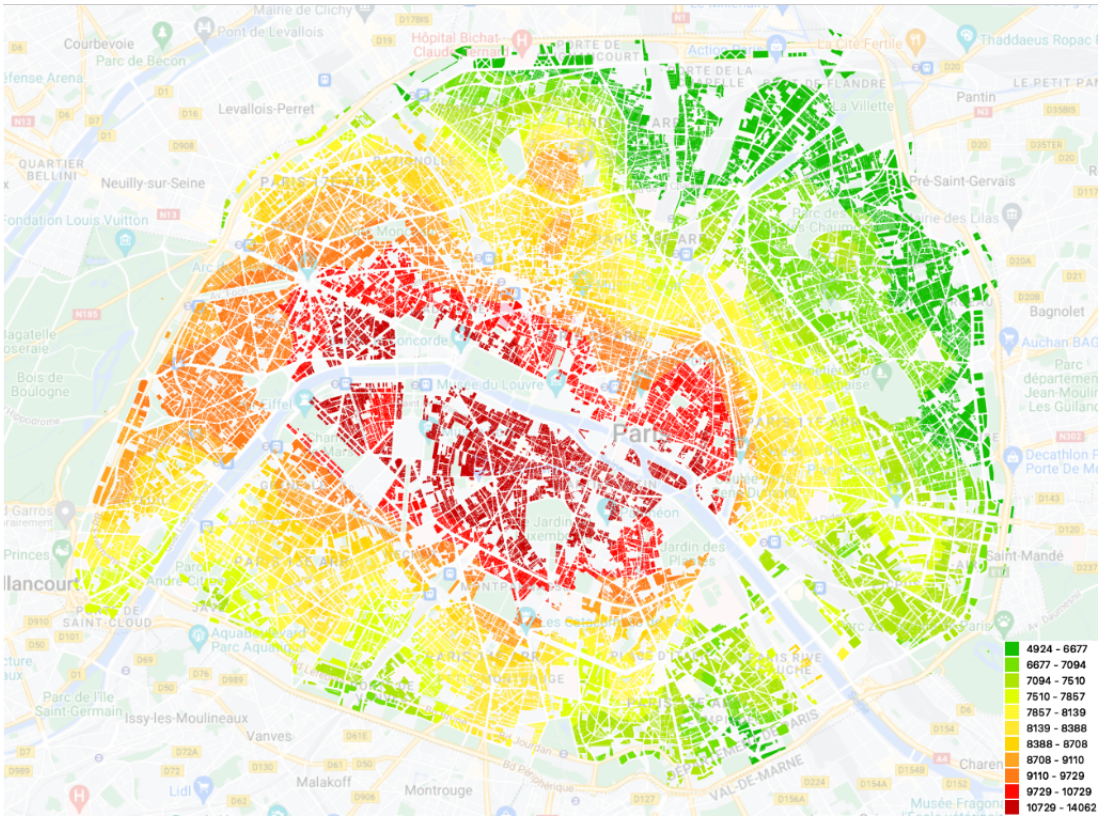


FIGURE 6.7 – État initial du réseau : carte des prix à Paris au 1er janvier 2016

Le modèle se montre relativement robuste à ces variantes d'initialisation, comme on peut le voir sur la figure 6.8, qui suit l'erreur médiane de prédiction, chaque mois, pour différents types d'initialisation du réseau de neurones.

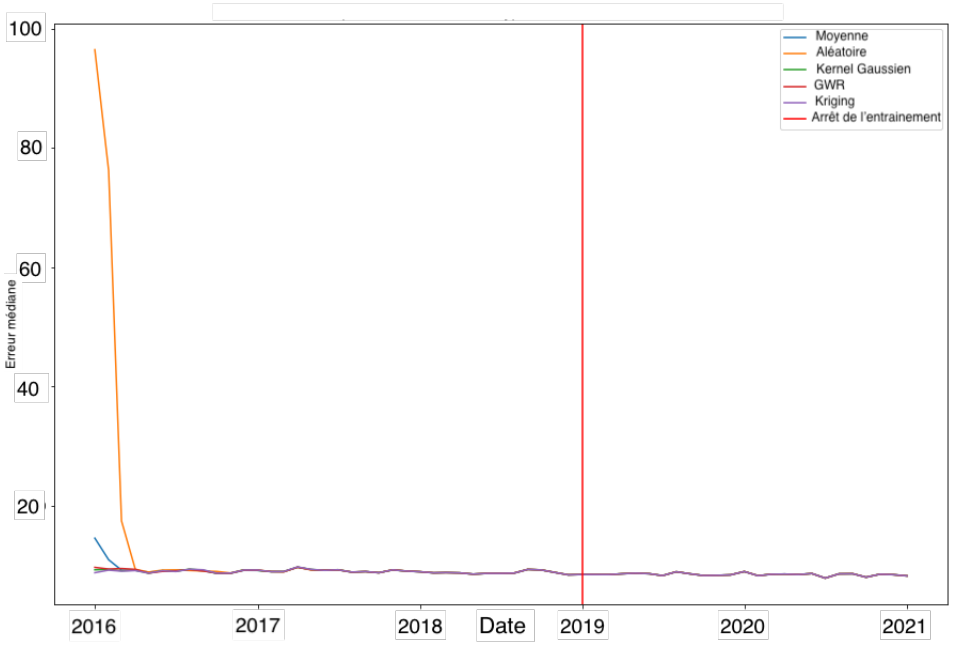


FIGURE 6.8 – Erreur médiane de prédiction mensuelle pour les réseaux de neurones avec des initialisations différentes, pour la ville de Paris (l'entraînement, jusqu'à 2018-12, fait référence à la période d'optimisation/apprentissage des paramètres α et σ, comme expliqué dans la section 6.3).

L'initialisation a un impact sur les trois premiers mois pour seulement deux types d'initialisations : Moyenne et Aléatoire (Figure 6.8) et est complètement neutre sur le reste de la vie du réseau de neurones. Même en partant d'une initialisation sans aucune structure spatiale (Aléatoire : erreur médiane à 98% au premier mois) le réseau s'adapte bien aux données qu'il reçoit en entrée chaque mois. Le choix de l'initialisation est donc négligeable.

6.3 Optimisation/Apprentissage des paramètres

On cherche à optimiser les paramètres σ et α qui sont au coeur du modèle et gouvernent la taille des quartiers et la manière dont l'information liée à une nouvelle transaction impacte la mise à jour du prix au niveau d'une parcelle.

6.3.1 Optimisation par *grid search*

Nous pouvons d'abord procéder suivant le protocole général mis en œuvre aux chapitres 4 et 5 : le modèle est mis à jour chaque mois sur la période T_{opt}=[2016-

01,2018-12] pour chaque couple (σ, α) appartenant à une grille $I_\sigma \times I_\alpha$. On calcule les erreurs de prédiction pour chacun de ces couples et on retient le couple qui minimise l'erreur médiane sur T_{opt}. Il y a au total $28 \times 29 = 812$ couples testés. Les erreurs médianes de prédiction à Paris sont affichées figure 6.9.

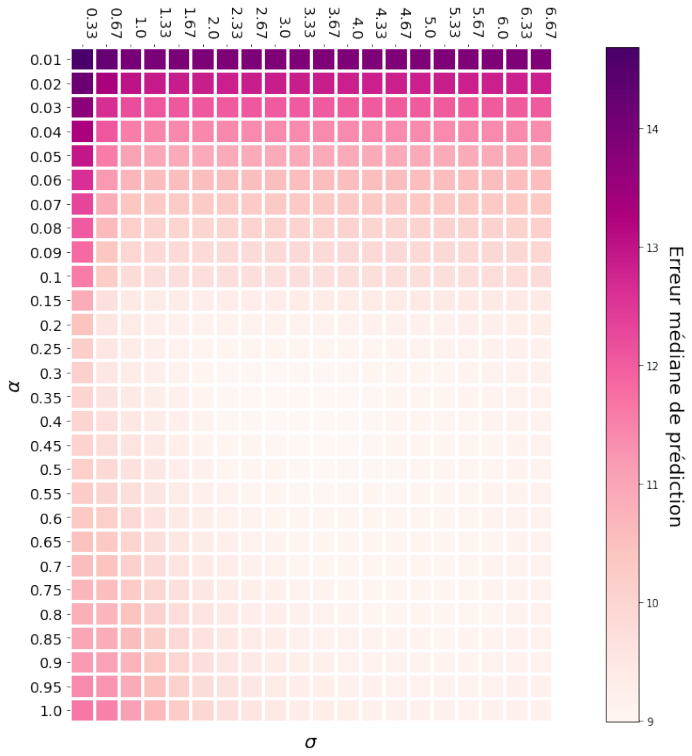


FIGURE 6.9 – Erreurs médianes de prédiction sur T_{opt}=[2016-01,2018-12] à Paris.

.

Le couple qui minimise l'erreur médiane est $\alpha = 0.4$, $\sigma = 3.6$. Cependant, cette démarche d'optimisation présente un inconvénient de taille pour son implémentation opérationnelle : elle nécessite environ 24 heures de calcul (sans optimisation experte du code)... Nous examinons donc d'autres démarches, d'abord en fixant arbitrairement un des deux paramètres et en « apprenant » l'autre à partir d'un critère de minimisation d'erreur, puis en apprenant les deux simultanément.

6.3.2 Un paramètre fixé, apprentissage de l'autre

Principe d'apprentissage du paramètre σ

Nous fixons ici la valeur de α et cherchons à apprendre la meilleure valeur de σ. Pour ce faire, plutôt que de travailler avec l'erreur médiane, moins explicite analytiquement, nous travaillons avec l'erreur quadratique moyenne **MSE**, définie par :

$$\mathbf{MSE}(\sigma) = \frac{1}{n(t+1)} \sum_{i=1}^{n(t+1)} (y_i - P_i(t+1))^2, \tag{6.6}$$

où $n(t+1)$ désigne le nombre de neurones i mis à jour à l'instant $t+1$ tels que $T_{N_i}(t) \neq \emptyset$, et y_i le prix de la transaction se situant sur le neurone i.

Le principe d'apprentissage consiste à déterminer σ^* tel que :

$$\sigma^* = \text{argmin}_\sigma \mathbf{MSE}(\sigma) \tag{6.7}$$

puis à mettre à jour la valeur de σ utilisée dans le modèle au temps $t+1$ par :

$$\sigma^{t+1} = \sigma^t + \eta(\sigma^* - \sigma^t), \tag{6.8}$$

où η est un pas d'apprentissage.
Ici, σ^* est directement trouvé en résolvant :

$$\frac{\partial \mathbf{MSE}(\sigma^*)}{\partial \sigma^*} = 0 \tag{6.9}$$

En posant $g_i(x) = y_i - P_i(t+1)(x)$, la dérivée partielle en fonction de σ est donc :

$$\frac{\partial \mathbf{MSE}(\sigma)}{\partial \sigma} = \frac{2}{n(t+1)} \sum_{i=1}^{n(t+1)} (y_i - P_i(t+1)(\sigma)) \left(g_i'(\sigma)\right), \tag{6.10}$$

avec :

$$g_i(\sigma) = y_i - P_i(t) - \alpha \frac{\sum\limits_{a \in T_{N_i}(t)} h_{ai}[P_a(t) - P_i(t)]}{\sum\limits_{a \in T_{N_i}(t)} h_{ai}}. \tag{6.11}$$

D'où :

$$g_i'(\sigma) = -\alpha \left(\frac{\sum_{a \in T_{N_i}(t)} h_{ai}'(\sigma)(P_a(t) - P_i(t))}{\sum_{a \in T_{N_i}(t)} h_{ai}} - \frac{\sum_{a \in T_{N_i}(t)} h_{ai}'(\sigma) \sum_{a \in T_{N_i}(t)} h_{ai}(\sigma)(P_a(t) - P_i(t))}{\left(\sum_{a \in T_{N_i}(t)} h_{ai} \right)^2} \right), \tag{6.12}$$

avec $h_{ai}'(\sigma) = \frac{\|r_a - r_i\|^2}{\sigma^3} h_{ai}(\sigma) = \frac{\|r_a - r_i\|^2}{\sigma^3} \exp\left(-\frac{\|r_a - r_i\|^2}{2\sigma^2(t)}\right)$.
On a finalement :

$$\frac{\partial \mathbf{MSE}(\sigma)}{\partial \sigma} = \frac{-2\alpha}{n(t+1)} \sum_{i=1}^{n(t+1)} (y_i - P_i(t+1)(\sigma)) \left(\frac{\sum_{a \in T_{N_i}(t)} h_{ai}'(\sigma)(P_a(t) - P_i(t))}{\sum_{a \in T_{N_i}(t)} h_{ai}} - \frac{\sum_{a \in T_{N_i}(t)} h_{ai}'(\sigma) \sum_{a \in T_{N_i}(t)} h_{ai}(\sigma)(P_a(t) - P_i(t))}{\left(\sum_{a \in T_{N_i}(t)} h_{ai} \right)^2} \right) \tag{6.13}$$

Nous examinons maintenant la même démarche sur l'autre paramètre du modèle, α.

Principe d'apprentissage du paramètre α

Comme pour σ, on cherche α^* tel que :

$$\alpha^* = \text{argmin}_\alpha \mathbf{MSE}(\alpha). \tag{6.14}$$

La règle de mise à jour de α au temps $t+1$ est :

$$\alpha^{t+1} = \alpha^t + \eta(\alpha^* - \alpha^t) \tag{6.15}$$

α^* est directement trouvé en résolvant :

$$\frac{\partial \mathbf{MSE}(\alpha^*)}{\partial \alpha^*} = 0, \tag{6.16}$$

soit :

$$\frac{-2}{n(t+1)} \sum_{i=1}^{n} (t+1)\,(y_i - P_i(t+1)) \frac{\sum\limits_{a \in T_{N_i}(t)} h_{ai}(P_a(t) - P_i(t))}{\sum\limits_{a \in T_{N_i}(t)} h_{ai}} = 0. \tag{6.17}$$

La fonction du pas d'apprentissage et sa valeur initiale ont été determinées par *grid search*. Parmi les fonctions existantes dans la littérature, voici celles qui ont été testées :

- Pas d'apprentissage basé sur le temps :

$$\eta_{\mathrm{TB}}^{t+1} = \frac{\eta^t}{1 + dn} \tag{6.18}$$

  où η est le pas d'apprentissage, n est le pas d'itération et d le paramètre de décroissance.

- Pas d'apprentissage tel qu'on peut le retrouver dans la littérature sur SOM [32] :

$$\eta_{\mathrm{SOM}}^{t+1} = \eta^0 \left(1 - \frac{n}{N}\right) \tag{6.19}$$

  où N est le nombre total d'itérations.

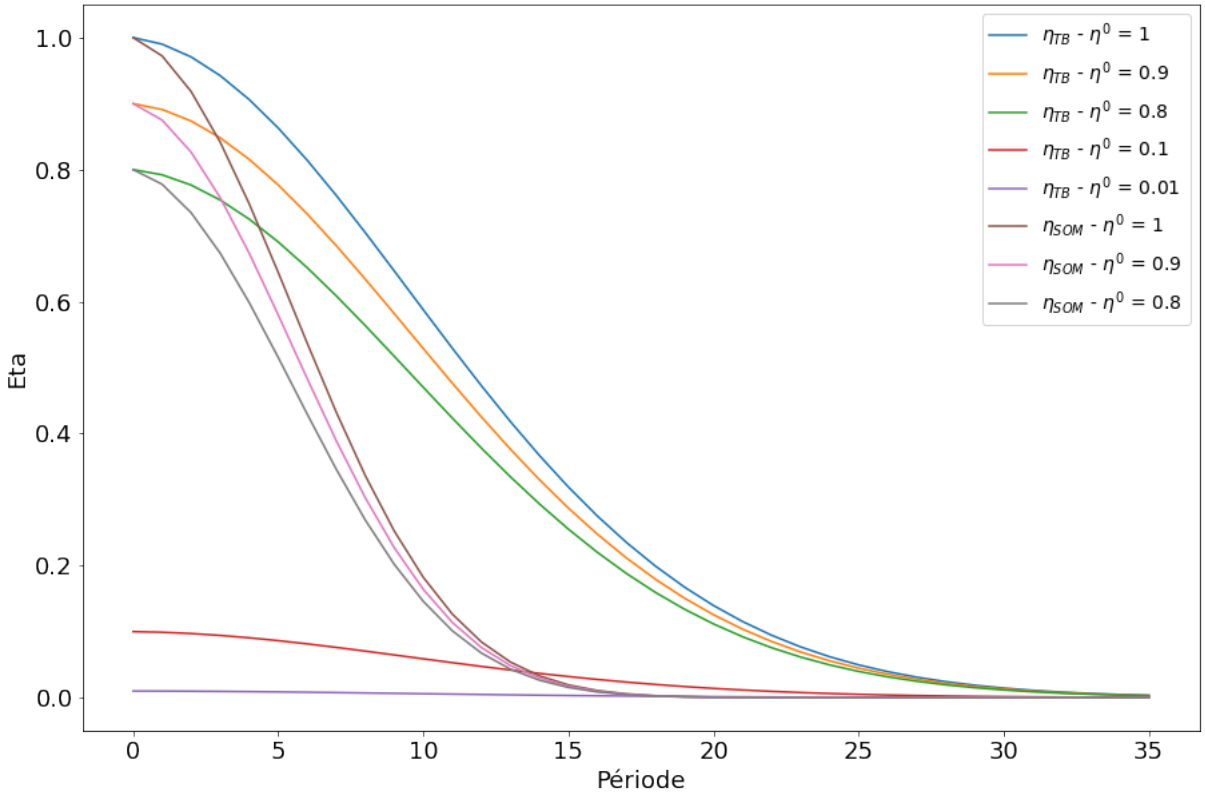


FIGURE 6.10 – Différentes fonctions de pas d'apprentissage

Comme précédemment, on retient la fonction du pas d'apprentissage qui minimise l'erreur de prédiction médiane sur la période de validation T_{opt}. Dans notre cas c'est la fonction se basant sur le temps (formule 6.18) avec $\eta^0 = 0.8$ comme le montre la figure 6.11.

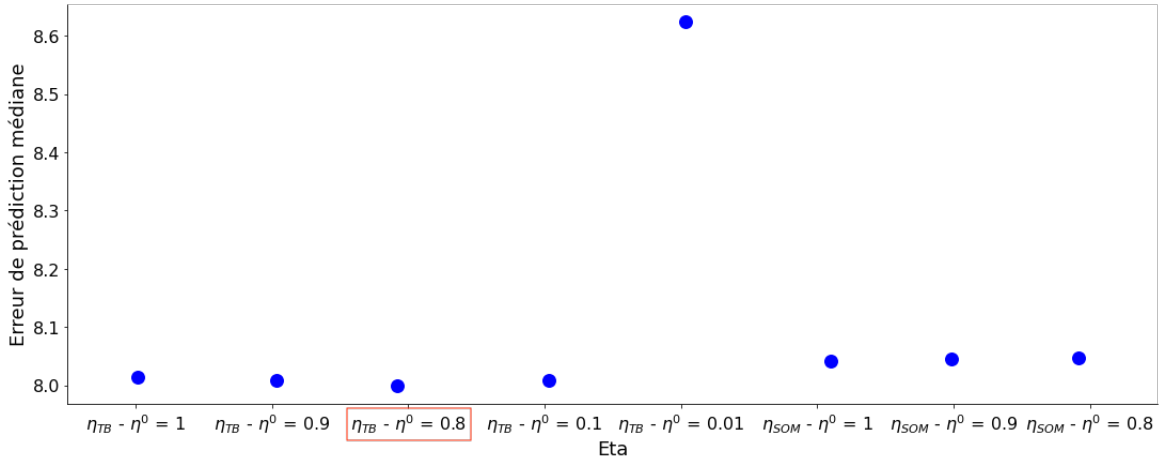


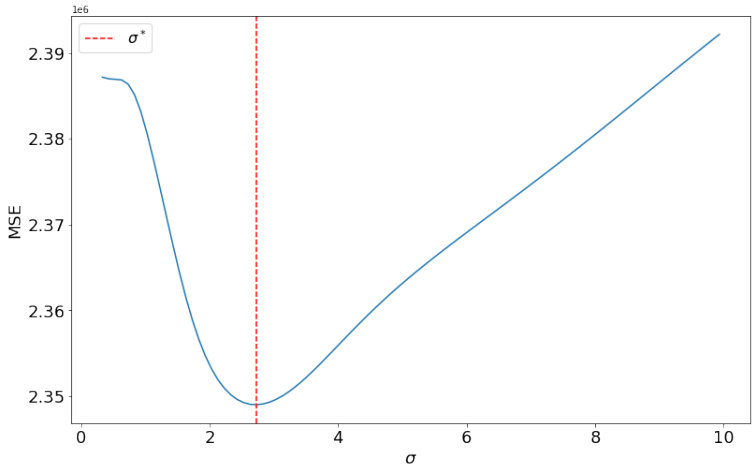
FIGURE 6.11 – Erreur de prédiction médiane sur la période T_{opt} pour les différentes fonctions de pas d'apprentissage.

6.3.3 Mis en œuvre de l'apprentissage

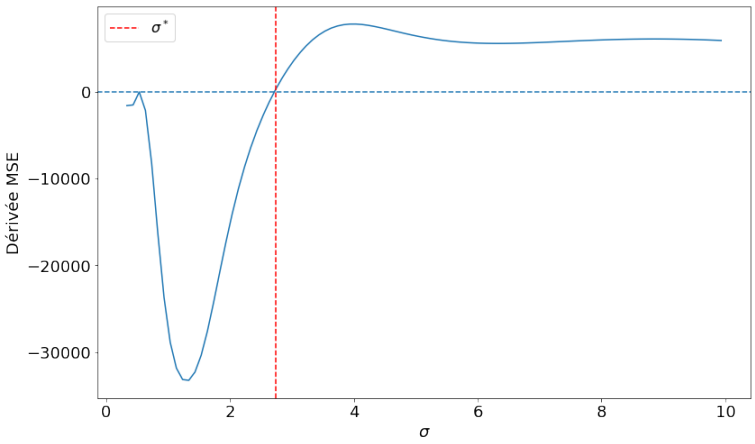
La période d'apprentissage T_{learn} est choisie identique à la période d'optimisation retenue pour les méthodes précédentes, T_{opt}=[2016-01,2018-12]. Les intervalles de recherche des paramètres sont fixés préalablement à $I_\sigma = [\frac{1}{3}, 10]$ et $I_\alpha = [0, 1]$ pour σ et α respectivement. Dans un premier temps, l'apprentissage des paramètres se fait pour la ville entière ; dans un second temps, on décide de subdiviser la ville afin d'apprendre les paramètres de manière locale. Dans tous les cas, les performances du modèle sont testées sur une période de deux ans, allant de 2019-01 à 2021-01, dénotée T_{val}.

Apprentissage de σ, α fixe

On cherche $\sigma^* \in I_\sigma$ pour une valeur de α fixe dans I_α. L'initialisation de σ se fait de manière aléatoire dans I_σ. La Figure 6.12 montre l'optimisation du paramètre σ pour le premier mois à Paris.



(a) MSE en fonction de σ dans l'intervalle de recherche I_σ



(b) Dérivée de la MSE en fonction de σ

FIGURE 6.12 – MSE et sa dérivée en fonction de σ pour $t = 1$ (2016-02) à Paris ($\text{Card}(T(t = 1)) = 1571$), avec $\alpha = 0.23$ choisi de manière aléatoire dans I_α. Le rayon optimal pour $t = 1$ est $\sigma^* = 2.7$

La valeur de σ^* est obtenue numériquement avec la fonction **brentq** du module **optimize** appartenant au package **scipy**. Elle consiste à trouver une racine d'une fonction dans un intervalle donné en utilisant la méthode de Brent.

En répétant la procédure chaque mois sur la période d'apprentissage T_{learn}, on obtient (pour Paris) les valeurs de σ^* et σ représentées sur la figure 6.13.

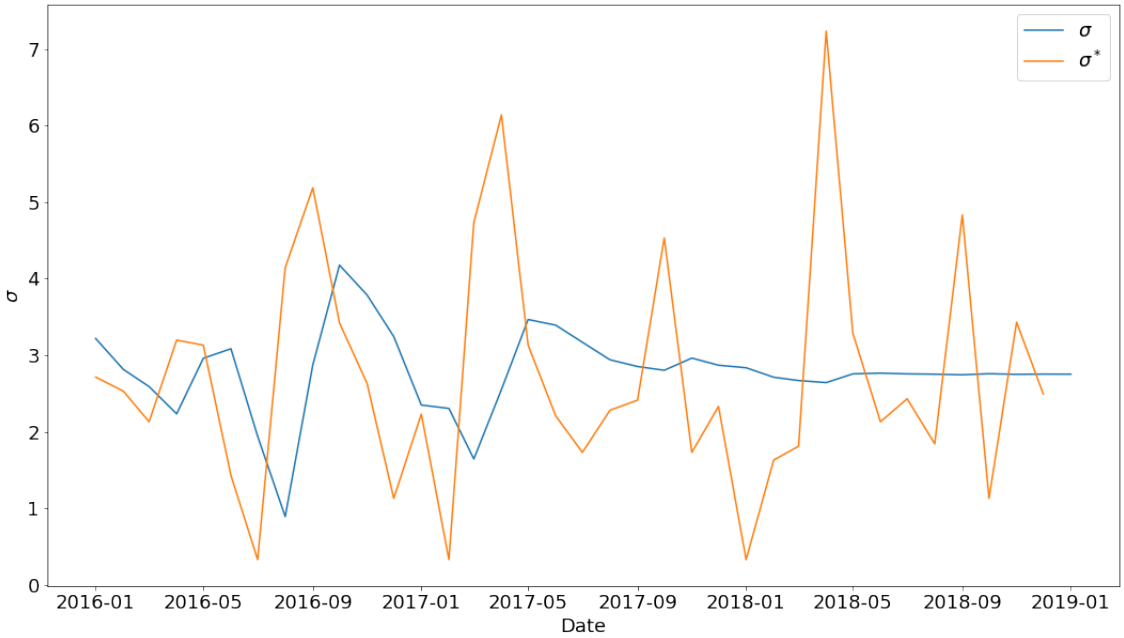


FIGURE 6.13 – Évolution de σ et σ^* sur T_{learn} pour un rayon initial $\sigma = 3.22$ et avec $\alpha = 0.23$ fixe, à Paris. On obtient un paramètre final $\sigma = 3$

.

Si l'on voit une stabilisation du rayon σ appris au fil du temps (figure 6.13) principalement due à la forme de la fonction de pas d'apprentissage η, on peut observer une oscillation du rayon optimal σ^*. Néanmoins l'intervalle dans lequel évolue σ^* pour tout t est raisonnable : $\sigma^* \in [0.33, 7.1]$.

La sensibilité à l'initialisation est mesurée en testant plusieurs rayons initiaux choisis aléatoirement. Un exemple à Paris est montré sur la figure 6.14.

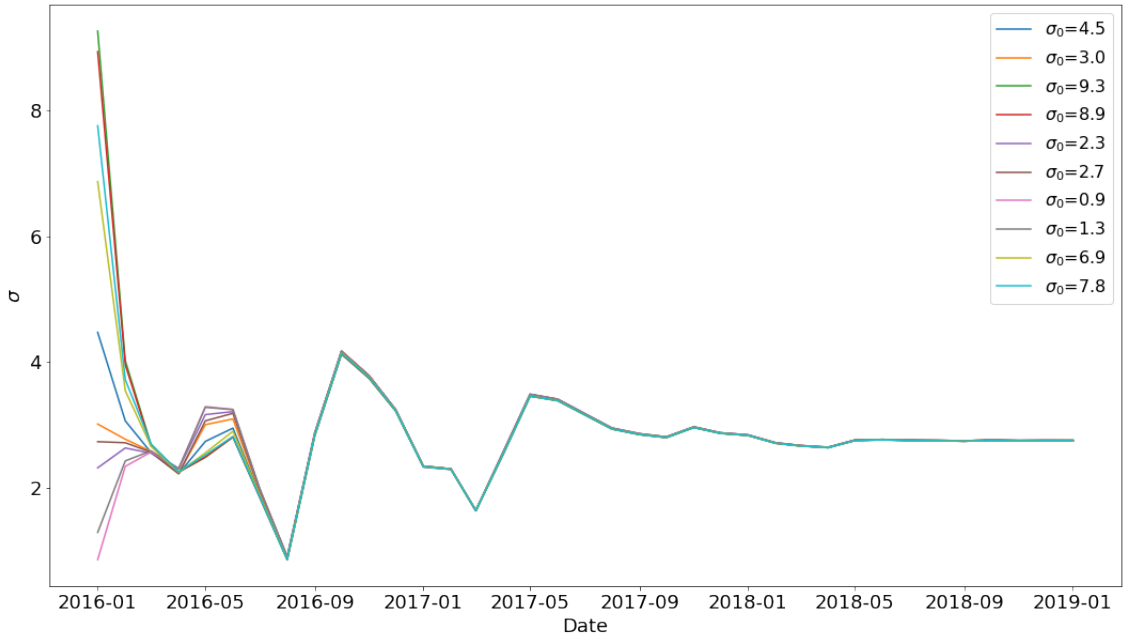


FIGURE 6.14 – σ sur I_{learn} pour 10 rayons initiaux σ_0 choisit aléatoirement dans I_σ.

.

Il faut 7 mois (2016-08) pour converger vers le même rayon σ (voir Figure 6.14).

Nous pouvons maintenant examiner les résultats avec la valeur apprise sur T_{learn} pour σ, que nous noterons σ' et qui vaut en l'occurrence 3. La figure 6.15 représente la médiane des erreurs relatives commises par le modèle à Paris, à la fois pendant la phase d'apprentissage (T_{learn}=[2016-01,2018-12]) et la phase de test (T_{val}=[2019-01,2021-01]).

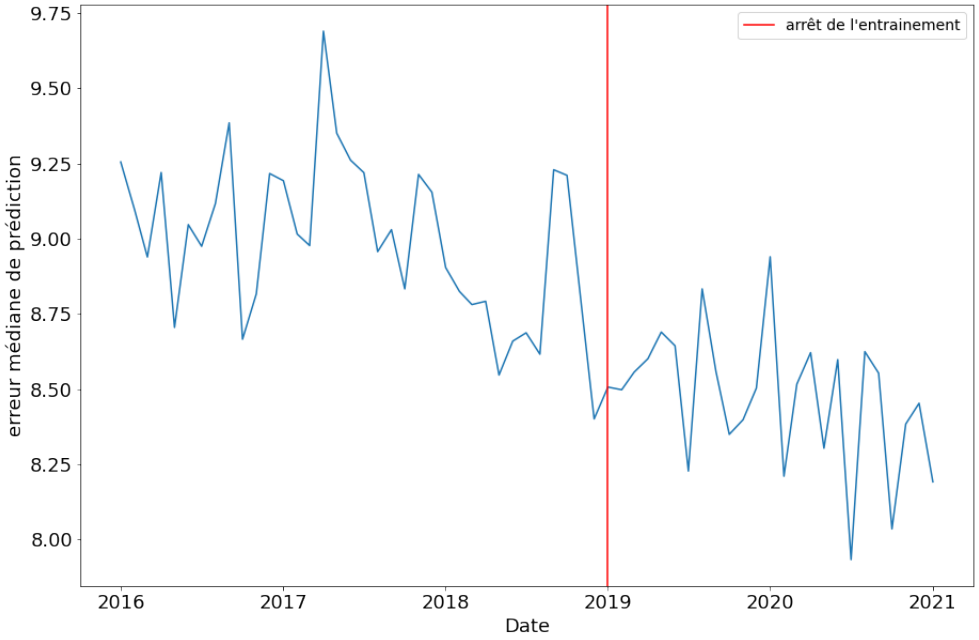


FIGURE 6.15 – Erreurs médianes de prédiction à Paris sur la période 2016-2021.

.

Afin de bien illustrer l'intérêt d'apprendre le paramètre σ, nous comparons les erreurs médianes de prédiction commises avec dix valeurs de σ différentes et fixes tout au long de la période $T_{\text{learn}} \cup T_{\text{val}}$ (et avec toujours le même $\alpha = 0.23$ fixe). La figure 6.16 représente l'ordre de classement par erreur médiane de prédiction mensuelle à Paris sur la période T_{val}.

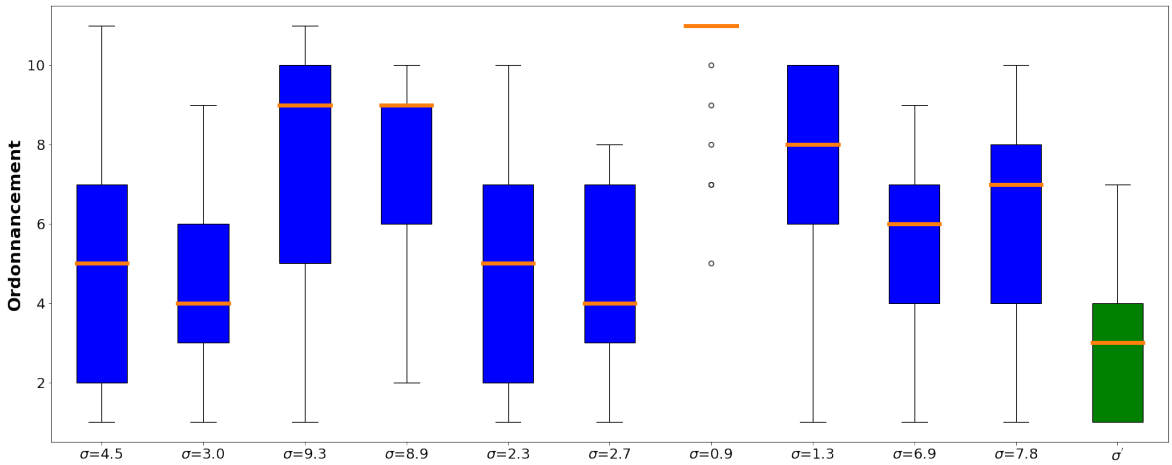


FIGURE 6.16 – Distributions de l'ordre de classement en fonction de l'erreur médiane mensuelle pour 10 valeurs de σ et pour σ appris (boite à moustache verte).

.

Dans 25% des cas, le modèle avec σ appris minimise l'erreur médiane de prédiction, et dans 50% des cas, il est au moins en troisième position.

Examinons à présent l'influence de la valeur de α, choisie arbitrairement et gardée fixe.

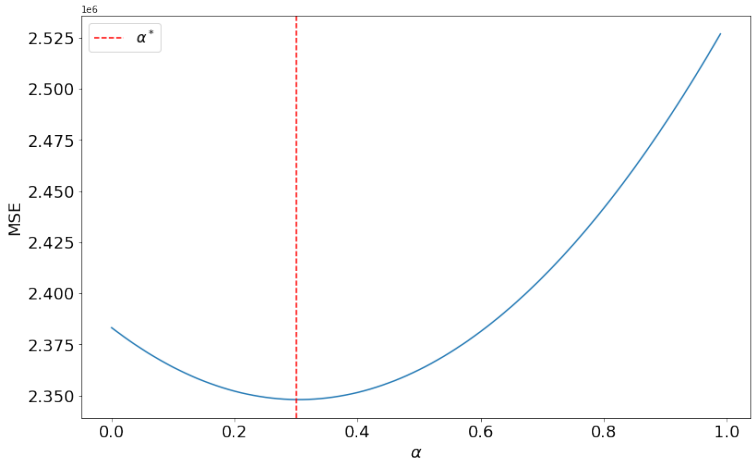
α	$\sigma^{'}$
0.3	3.1
0.5	3.7
0.9	4.6

TABLE 6.1 – $\sigma^{'}$ en fonction de α fixe, pour Paris

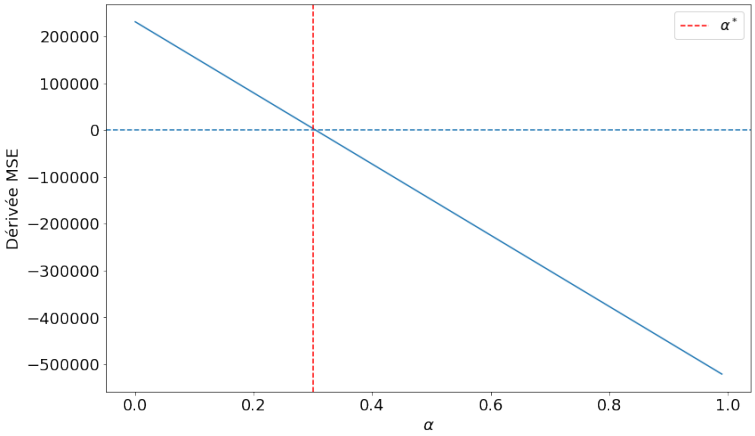
La valeur apprise de σ est directement liée à la valeur de α (voir tableau 6.1). En effet, si α a une valeur basse, alors le réseau n'a quasiment pas d'inertie et ne prend que très peu en compte les nouvelles transactions : σ n'a alors pas besoin d'être large. Au contraire, si α est grand (par exemple 0.9) alors le réseau a une inertie forte et σ, mécaniquement, doit augmenter pour éviter que le modèle ne fasse trop d'erreurs.

Apprentissage de α, σ fixe

Nous inversons maintenant les rôles entre les deux paramètres pour l'apprentissage. Nous cherchons $\alpha^* \in I_\alpha$ pour une valeur de σ fixe dans I_σ. L'initialisation de α à $t = 0$ se fait de manière aléatoire dans I_α. La figure 6.17 montre l'optimisation du paramètre α pour le premier mois ($t = 1$) à Paris.



(a) MSE en fonction de α dans l'intervalle de recherche I_α



(b) Dérivée de la MSE en fonction de α

FIGURE 6.17 – MSE et sa dérivée en fonction de α pour $t = 1$ (2016-02) à Paris ($\text{Card}(T(t = 1)) = 1571$), avec $\sigma = 3.22$ choisi de manière aléatoire dans I_σ. On obtient $\alpha^* = 0.27$ pour $t = 1$.

En répétant la procédure chaque mois sur la période d'apprentissage T_{learn}, on obtient (pour Paris) les valeurs de α et α^* représentées sur la figure 6.18.

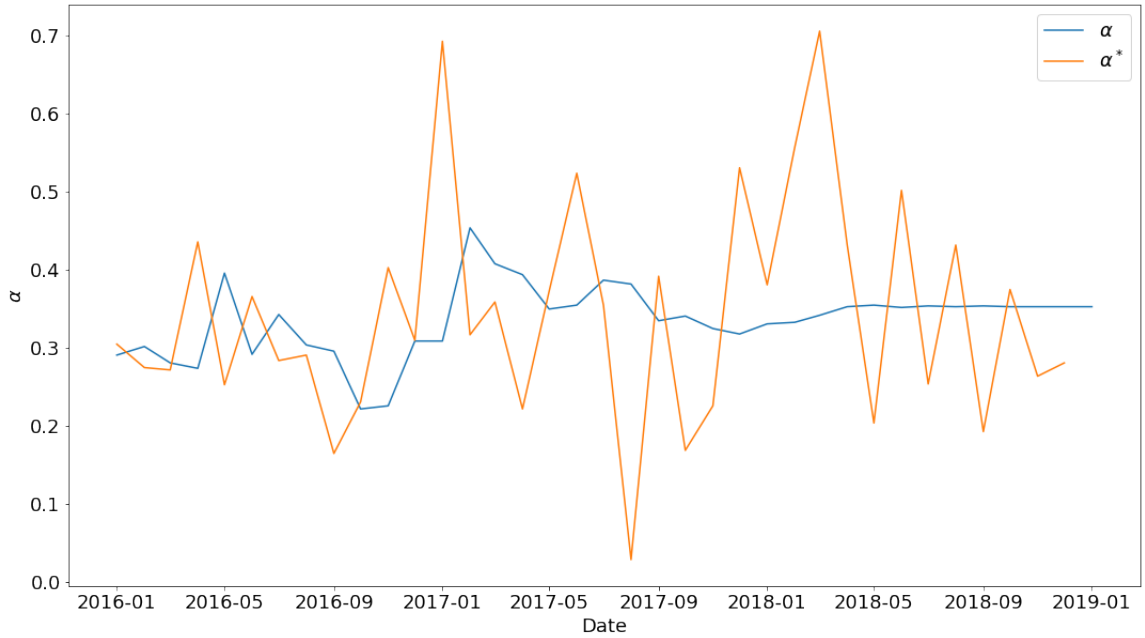


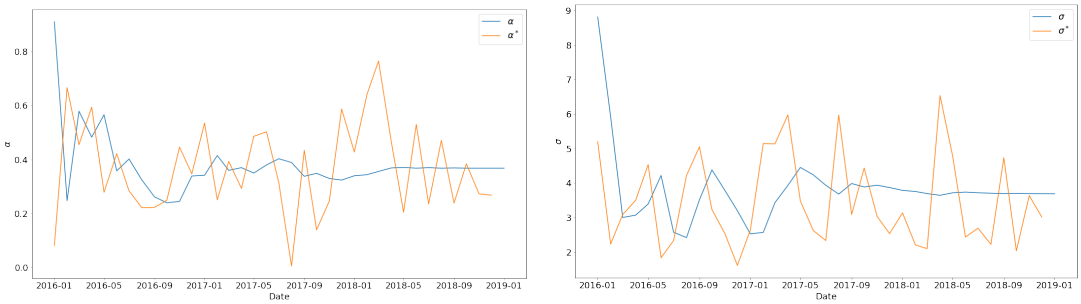
FIGURE 6.18 – Évolution de α et α^* sur T_{learn} pour une initialisation aléatoire de $\alpha = 0.29$ et avec $\sigma = 3.22$ fixe, à Paris. On obtient un paramètre final $\alpha = 0.35$

.

Comme dans le cas de σ, on constate une stabilisation de α liée à la forme de la fonction d'apprentissage. La valeur apprise ici au final est $\alpha^{'} = 0.35$.

Apprentissage conjoint de α, σ

On cherche désormais conjointement $\alpha^* \in I_\alpha$, $\sigma^* \in I_\sigma$ chaque mois, avec une initialisation aléatoire des paramètres α et σ dans I_α et I_σ respectivement. Nous utilisons ici à chaque pas de temps la même recherche de racines des dérivées partielles que dans les deux paragraphes précédents, mais en le faisant à chaque fois pour les deux paramètres, et en mettant ainsi à jour à chaque pas de temps leurs valeurs.



(a) α et α^* lors de la période d'entrainement (36 mois) à Paris.

(b) σ et σ^* lors de la période d'entrainement (36 mois) à Paris.

FIGURE 6.19 – Évolution des valeurs de α et σ pour une initialisation aléatoire égale à $\alpha = 0.91$ et $\sigma = 8.81$. On obtient des paramètres finaux égaux à $\alpha' = 0.36$ et $\sigma' = 3.6$ ce qui est très proche des paramètres obtenus pour les optimisations partielles (voir fig. 6.13 et 6.18).

Les valeurs obtenues des paramètres ($\alpha' = 0.36$ et $\sigma' = 3.6$) sont très similaires à ceux obtenus par *grid search* tout au début de cette section. Néanmoins, il y a un réel intérêt à utiliser l'approche par apprentissage du point de vue du temps de calcul : 3 heures et demie au lieu de 24 heures...

Illustrons à nouveau l'impact de l'apprentissage des paramètres α et σ. La figure 6.20 représente les distributions des erreurs de prédiction sur la période T_{val}, à Paris, pour des choix arbitraires de α et σ, comparés aux performances du modèles avec apprentissage.

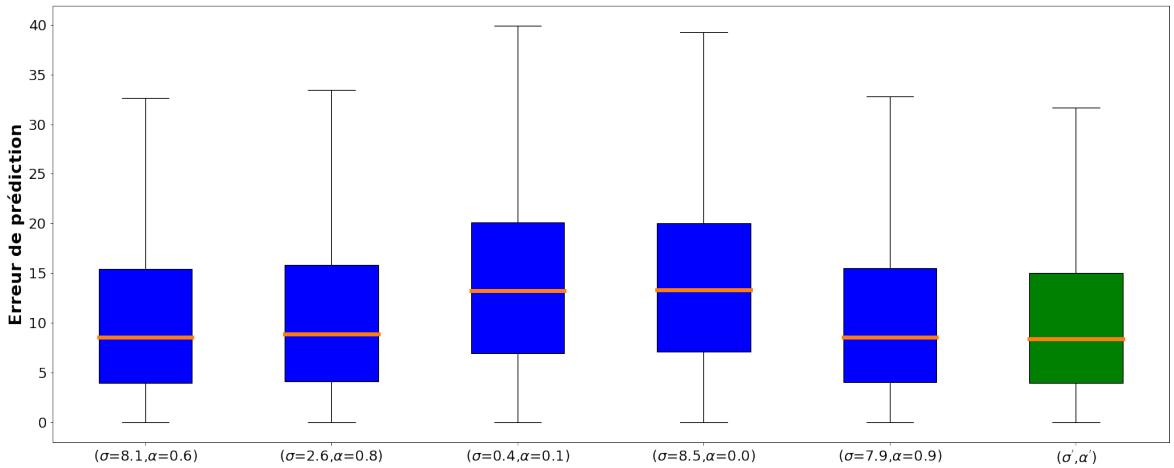


FIGURE 6.20 – Distribution des erreurs de prédiction sur la période de validation à Paris pour six couples de paramètres (σ,α) fixes arbitraires. On compare ces erreurs à celles du modèle avec l'apprentissage.

En regardant l'erreur médiane de prédiction (figure 6.20), on voit qu'il est nécessaire d'optimiser les paramètres (ce qui parait assez naturel). En effet, des paramètres pris aléatoirement et n'ayant pas beaucoup de sens (par exemple $\sigma = 0.4$, $\alpha = 0.1$) mènent à des erreurs de prédiction largement au dessus des modèles optimisés (erreur médiane à 14).

On veut maintenant analyser les erreurs de prédiction sur des modèles ayant différents types d'apprentissages. On cherche à savoir s'il y a une différence entre une optimisation jointe et une optimisation partielle. Pour cela, on compare les erreurs de prédiction pour les trois approches. Le cas à Paris est décrit sur la figure 6.21.

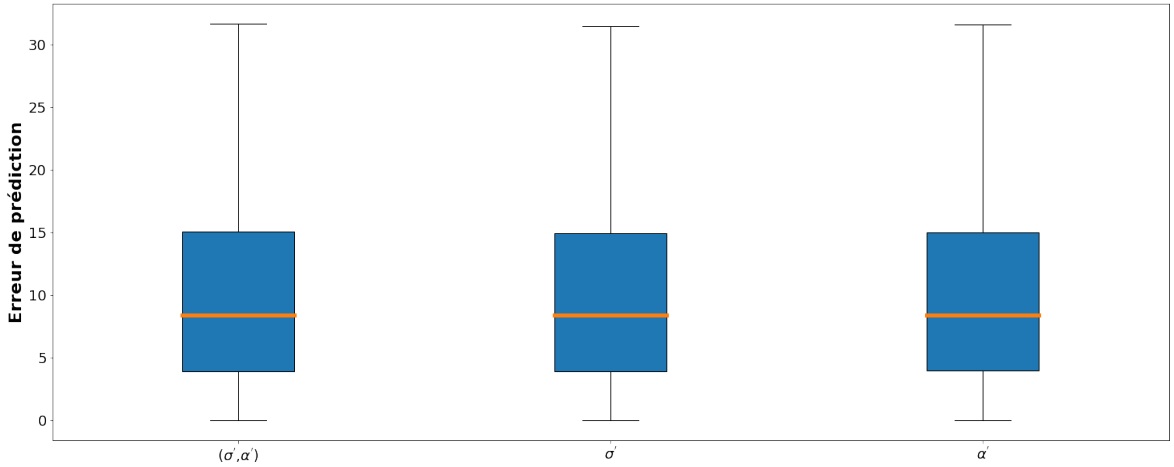


FIGURE 6.21 – Distribution des erreurs de prédiction pour des optimisations jointes ou partielles sur la période de validation T_{val} à Paris

.

Si on prend en compte uniquement l'erreur médiane des trois modèles (Figure 6.21) on peut penser qu'il n'y a pas grand intérêt à optimiser le couple de paramètres, car cela donne le même niveau d'erreur que pour une optimisation partielle. En effet, les paramètres α et σ sont liés et l'un compense l'autre. Par exemple, si α est grand, c'est-à-dire, une très faible mémoire du passé, alors σ va s'adapter en prenant de plus grandes valeurs, pour composer la trop forte réactivité de α. Cependant, les paramètres laissés fixes et pris aléatoirement n'ont pas des valeurs extrêmes pour notre exemple (6.17 et 6.12). Il faudrait mesurer les erreurs de prédictions pour des valeurs extrêmes des paramètres fixes et voir comment se comporte le paramètre optimisé. Dans tous les cas, le coût supplémentaire (en tant de calcul) d'une optimi-

sation jointe par rapport à une optimisation partielle est négligeable même lorsqu'on fait face à beaucoup de données. Dans la suite on considère uniquement le cas d'une optimisation jointe.

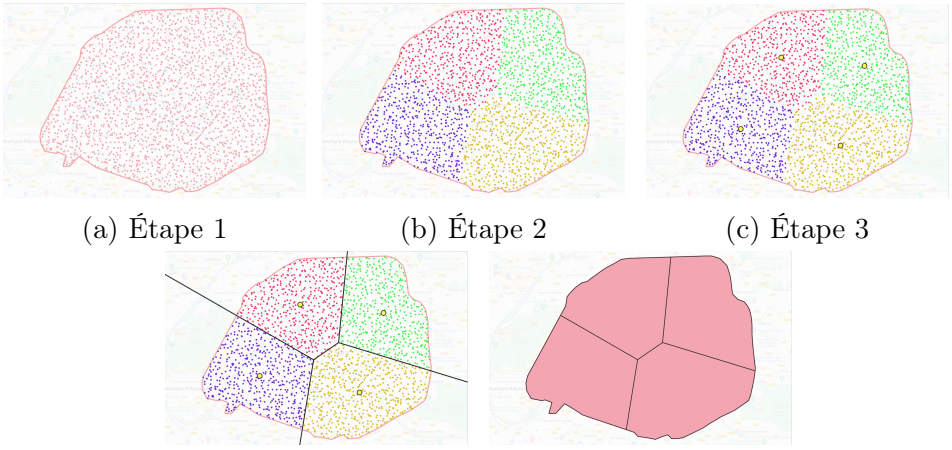
Apprentissage local et ultra-local

On décide dans cette section d'apprendre les paramètres du modèle de manière locale voir ultra-locale. L'idée sous-jacente est que le processus de diffusion de l'information n'est pas homogène sur toute une ville et de ce fait, les paramètres σ et α n'auront pas nécessairement la même valeur selon les localisations. En effet, un quartier en plein embourgeoisement où les prix peuvent augmenter de manière rapide dans un périmètre bien défini n'a pas la même dynamique qu'un quartier bourgeois ancien où les prix sont élevés depuis longtemps.

Afin de diviser la ville en plusieurs parties, on utilise la solution proposée par Darafei Praliaskouski, développeur PostGIS. Le processus est le suivant :

1. création d'un champ de points qui remplit le polygone (ie la géométrie de la ville), les points générés de manière aléatoire occupent une place homogène sur le polygone.
2. regroupement le champs de points en définissant le nombre de clusters en fonction du nombre de parties souhaitées pour diviser la ville.
3. calcul du centroïde de chaque cluster.
4. utilisation d'un diagramme de Voronoï afin d'obtenir des frontières entre les centroïdes des clusters.
5. intersection des régions de Voronoï avec le polygone d'origine pour obtenir les polygones finaux qui incluent à la fois les bords extérieurs du polygone d'origine et les frontières de Voronoï

Les différentes étapes énumérées ci-dessus sont représentées sur la figure 6.22 avec un exemple à Paris.



(a) Étape 1 (b) Étape 2 (c) Étape 3

(d) Étape 4 (e) Étape 5

FIGURE 6.22 – Exemple de division en 4 parties homogènes à Paris

Optimisation à partir des paramètres de krigeage

Dans cette partie on utilise les paramètres du krigeage, précisément ceux du variogramme, afin d'estimer le paramètre σ. Le variogramme borné est croissant jusqu'à un certain palier. On appelle la valeur de la distance où le variogramme se rapproche de son asymptote la portée. Les paires d'observations séparées par une distance inférieure à la portée sont autocorrélées spatialement alors qu'au-delà de cette distance elles ne le sont plus.

On décide alors d'utiliser la valeur du palier (en mètres) pour estimer le paramètre σ (sur le graphe) en se référant aux correspondances des distances calculées plus tôt (figure 6.5).

Le troisième paramètre du variogramme est la pépite. La pépite correspond à la valeur du variogramme pour une distance très proche de 0. Par définition, le variogramme en zéro vaut zéro, mais en pratique il existe des discontinuités à l'origine. Dans notre cas, deux appartements situés dans le même immeuble et présentant des caractéristiques exactement identiques n'ont pas le même prix (voir 6.23.

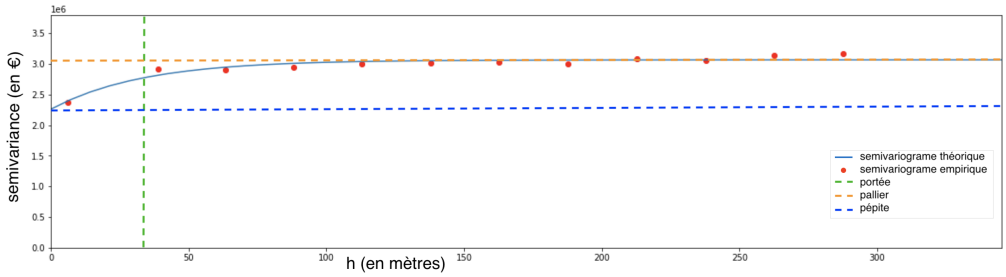


FIGURE 6.23 – Variogrammes empirique et théorique à Paris au 1er janvier 2021. Les paramètres sont les suivants : portée = 33, palier = 3064672, pépite = 2261080.
.

Pour notre exemple à Paris, on obtient un rayon sur le graphe égal à σ=1.13.

6.4 Résultats

6.4.1 Comparaison avec l'existant

Afin de mettre en relief nos résultats, nous les comparons avec des modèles largement utilisés dans la littérature à savoir la GWR (Geographically Weighted Regression) et le krigeage, et présentés au chapitre 1 puis avec les développements de ces modèles proposés dans les chapitres 4 et 5 de cette thèse.

Comme précédemment et comme décrit dans le protocole général (chap. 3), les prix sont prédits chaque premier mois sur la période $(T_{\text{learn}} + T_{\text{val}})$ et la performance du modèle est mesurée en calculant l'erreur de prédiction relative sur les transactions observées dans le mois.

6.4.2 Application des méthodes

La description du parc des villes de Paris et Les Lilas, et celle de leur réseau de neurones sont résumés dans les tableaux 6.2 et 6.3 respectivement.

TABLE 6.2 – Description du parc

	Paris	Les Lilas
Population	2220445	22762
% d'appartements	99	89
Nombre de logements (en millier)	1330.026	11.133
Densité de population (en km^2)	21010	18759
Nombre de parcelles	77218	1880
Nombre de parcelles résidentielles	66295	1644

		Paris	Les Lilas
Nombre neurones		122823	2157
Nombre neurones résidentiels		66295	1644
[2016-01-01,2018-12-01]	% neurones activés	39.6	15.1
	nombre observations moyen (par mois)	1786	16.1
[2019-01-01,2021-01-01]	% neurones activés	31	12.1
	nombre observations total	38630	392

TABLE 6.3 – Description du réseau de neurones. Le pourcentage de neurones activés représente le nombre de neurones où il y a eu une ou plusieurs transactions divisé par le nombre de neurones résidentiels. Le nombre de neurones activés indirectement dépend lui du paramètre σ.

Notre modèle est initialisé à 2016-01, les prix sont mis à jour chaque mois grâce aux nouvelles transactions. On représente ici l'état du réseau 5 ans après l'initialisation, c'est-à-dire à 2021-01.

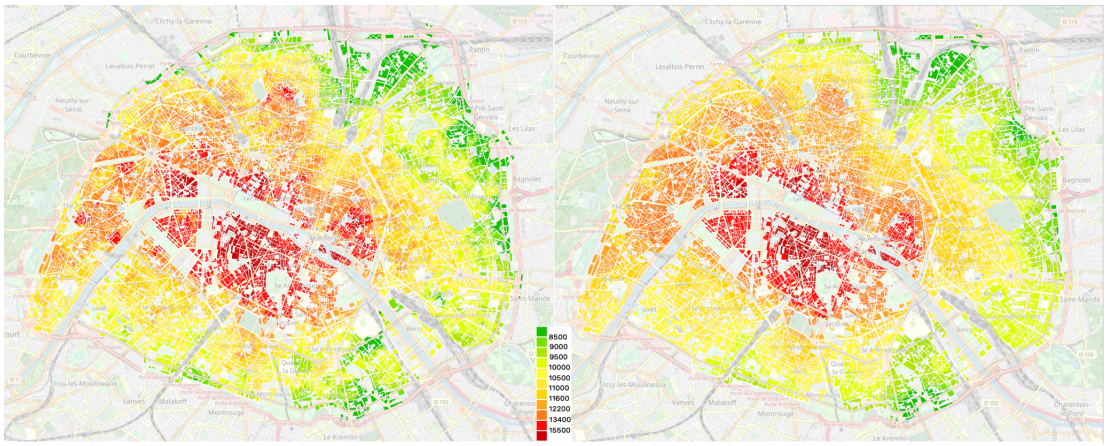


FIGURE 6.24 – Cartes des prix à Paris au 1er janvier 2021. À gauche : modèle avec σ fixé à partir de la portée obtenue avec le variogramme et optimisation de $\alpha \in I_\alpha$, on obtient $\sigma = 1.13$ et $\alpha' = 0.19$. À droite : modèle avec optimisation de $\sigma \in I_\sigma$ et $\alpha \in I_\alpha$ globalement sur toute la ville, on obtient $\sigma' = 3.7$, $\alpha' = 0.37$.

.

Comme on peut voir sur la figure 6.24, le fait d'utiliser des rayons différents sur le graphe produit une organisation spatiale globale de la ville similaire. Seulement, si on regarde de plus près, le fait d'utiliser un σ plus faible (Figure 6.24 gauche) mène à produire des quartiers plus resserrés comme autour du quartier de Montmartre par exemple ou bien de part et d'autre de l'avenue des Champs-Élysées.

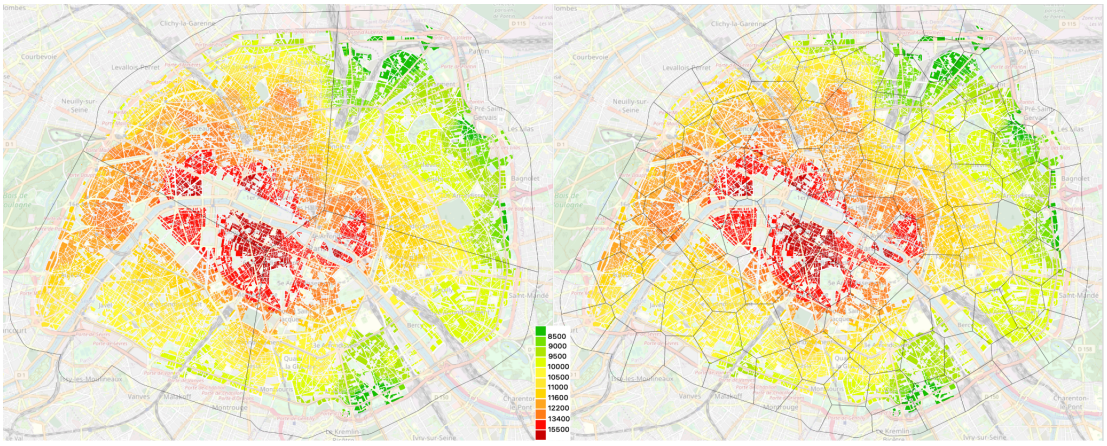


FIGURE 6.25 – Cartes des prix à Paris au 1er janvier 2021. À gauche : modèle avec optimisation de $\sigma \in I_\sigma$ et $\alpha \in I_\alpha$ pour chacune des 4 parties de la ville. À droite : modèle avec apprentissage de $\sigma \in I_\sigma$ et $\alpha \in I_\alpha$ pour chacune des 100 parties de la ville.

.

Appliquer un apprentissage local ou ultra-local mène à des résultats très similaires du point de vue de la formation des prix. Les cartes des prix présentes dans la figure 6.25 se ressemblent et pourtant celle de gauche bénéficie de 4 couples (α', σ') alors que celle de droite en utilise 100. Même si la distribution des paramètres (6.26) est plus étendue pour une optimisation sur 100 parties, la tendance générale est déjà entièrement captée par la division de la ville en 4 parties qui, à Paris, est caractéristique d'une typologie de la ville. L'ouest parisien bourgeois et homogène comparé à l'est historiquement plus populaire, mais morcelé par une gentrification progressive de ces quartiers. En effet, dans notre exemple les deux plus petits rayons sont ceux de l'est alors que les plus grands sont ceux de l'ouest. La figure 6.29 nous confirme l'impact négligeable d'une optimisation ultra-locale, car on ne voit aucune différence sur l'erreur de prédiction sur les ensembles de tests pour chacun des modèles.

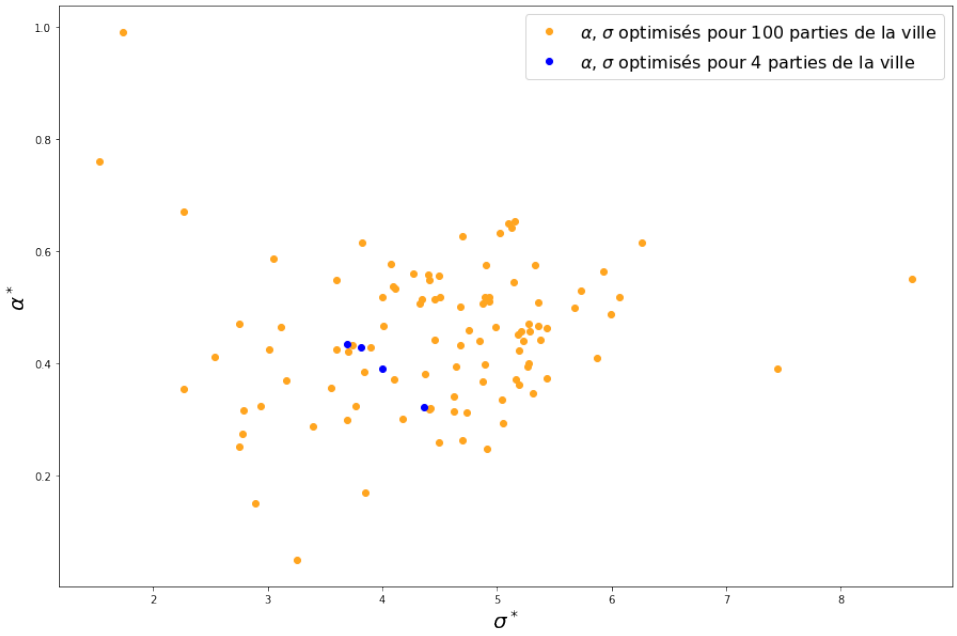
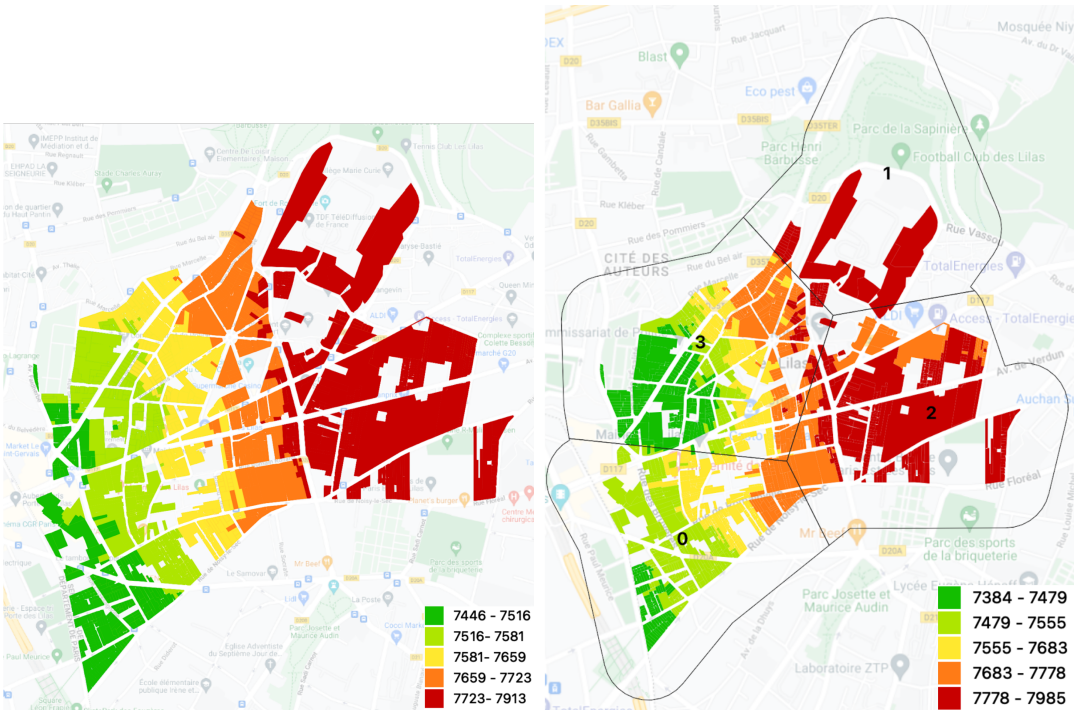


FIGURE 6.26 – Couples (α', σ') pour les deux modèles : découpage de la ville en 4 et en 100 parties

.

Les résultats du modèle appliqué sur la ville des Lilas sont présentés dans la Figure 6.27.



(a) Carte des prix au 1er janvier 2021 aux Lilas obtenue avec apprentissage des paramètres $\sigma \in I_\sigma$ et $\alpha \in I_\alpha$ sur toute la ville. On obtient $\sigma' = 6.5$ et $\alpha' = 0.56$

(b) Carte des prix au 1er janvier 2021 aux Lilas obtenue avec apprentissage des paramètres $\sigma \in I_\sigma$ et $\alpha \in I_\alpha$ pour chacune des 4 parties de la ville. On obtient σ'=(6.2,5.9,4.0,4.7) et α'=(0.64,0.8,0.4,0.53)

FIGURE 6.27 – Carte des prix aux Lilas au 1er janvier 2021

De la même manière qu'au chapitre 4, la carte produite par un apprentissage global sur toute la ville (Figure 6.27(a))) ne reflète pas vraiment une réalité de la structure des prix présente dans la ville des Lilas. De plus, la variance au sein des prix est très faible (de 7446€à 7913€). On peut noter que la structure spatiale change un peu en passant à un apprentissage local (Figure 6.27(b)), notamment pour les parcelles appartenant à la cellule 3. Celle-ci à un rayon optimisé $\sigma' = 4$ et donc dessine des quartiers plus fins.

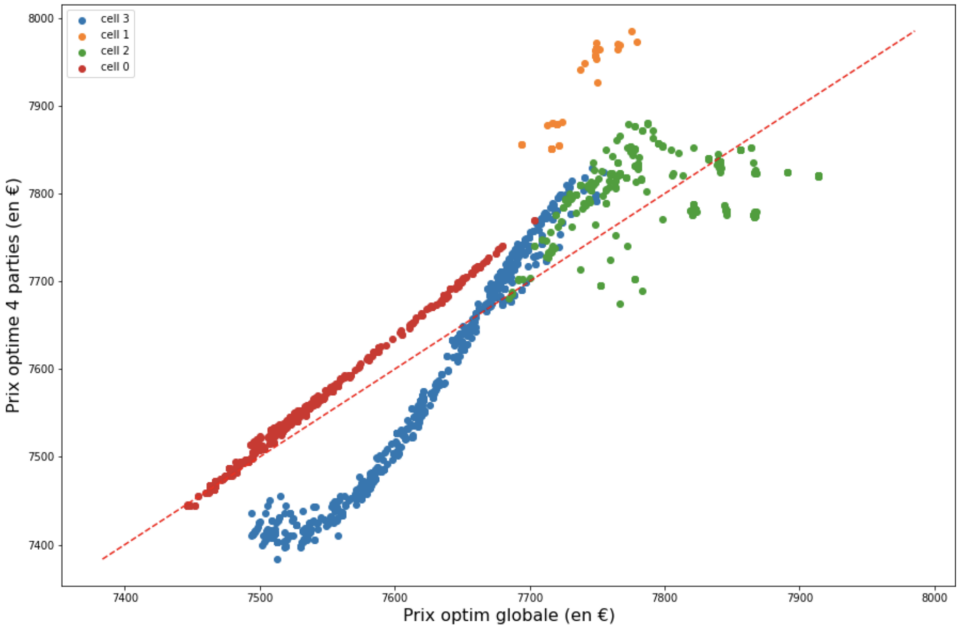


FIGURE 6.28 – Diagramme de dispersion avec en abscisse les prix obtenus avec un apprentissage des paramètres global et en ordonnée un apprentissage local (ville divisée en 4 parties) aux Lilas.

L'effet de l'apprentissage local peut être analysé à partir de la Figure 6.28. Les prix des parcelles situées sur la cellule 0 ne varient presque pas, car le rayon σ' appris localement est très semblable au rayon appris de manière globale (6.2 vs 6.5). On peut voir que les prix des parcelles appartenant à la cellule 1 augmentent avec un apprentissage local, le rayon étant plus faible ($\sigma' = 5.9$) et les transactions plus élevées à cet endroit, les prix augmentent naturellement, même si cette hausse reste très légère.

6.4.3 Performances comparées des modèles

La performance du modèle est mesurée en calculant l'erreur de prédiction relative par rapport aux transactions observées chaque mois. La figure 6.29 représente les distributions des erreurs de prédiction pour différentes partitions de la ville. On cherche à savoir si l'apprentissage local a un effet sur les performances.

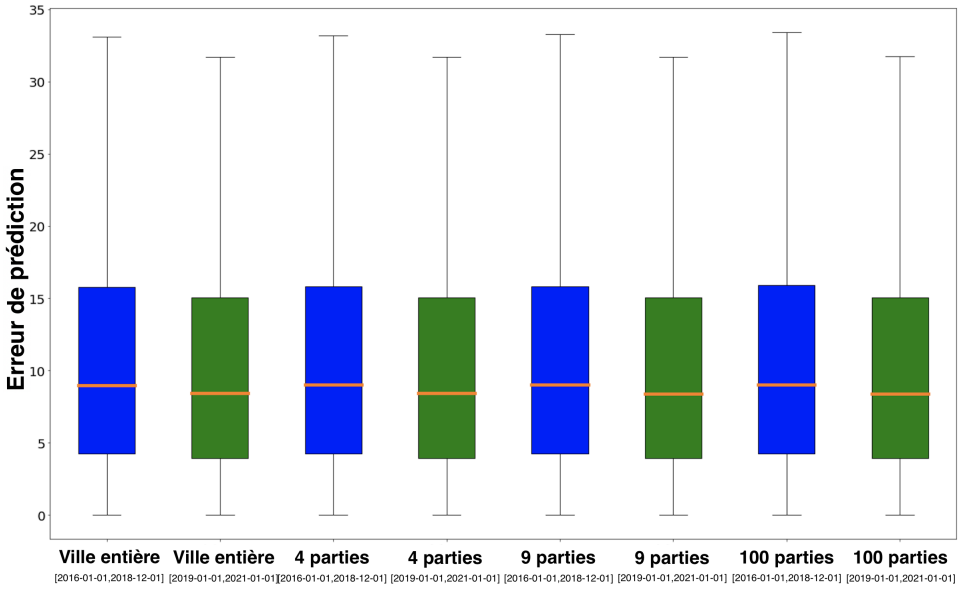


FIGURE 6.29 – Boîtes à moustache des erreurs de prédiction pour la ville de Paris. Le libellé **Ville entière** correspond au modèle avec un apprentissage de α et σ sur la ville entière. Le label **4 parties** correspond au modèle avec un apprentissage de α et σ sur chacune des 4 parties de la ville, etc.

On différencie l'erreur de prédiction sur la période T_{learn} où les paramètres σ et α sont appris (boîtes à moustache bleues) et la période T_{val} (boîtes à moustache vertes). Les volumes cumulés des transactions sur la période T_{learn} sont de 64216 et 581 à Paris et aux Lilas respectivement, ceux sur la période T_{val} sont de 40075 et 392 à Paris et aux Lilas respectivement.

Les erreurs de prédiction durant T_{learn} (boite à moustache bleue) sont plus élevées que celles durant la période de validation, T_{val} car à l'initialisation les paramètres σ et α sont choisis de façon aléatoire, ce qui peut entraîner de grosses erreurs de prédiction durant les premiers mois, le temps que le modèle s'ajuste.

L'apprentissage local, voire ultra-local, présentait très peu de différences sur les cartes de prix (figure 6.25), il n'y a pas non plus d'impact net sur l'erreur de prédiction, quelle que soit la granularité à laquelle les paramètres sont appris.

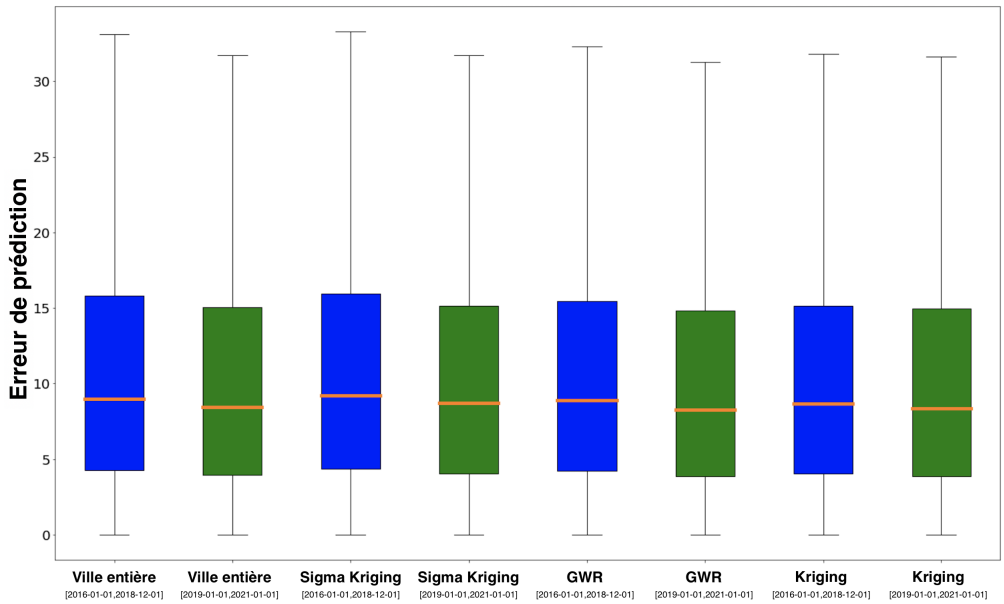


FIGURE 6.30 – Boîtes à moustache des erreurs de prédiction pour la ville de Paris. (Le modèle «Sigma Kriging» correspond à une mise à jour du réseau de neurones avec σ choisi par optimisation à partir des paramètres de krigeage et α appris sur T_{learn}.)

.

En comparant notre modèle aux modèles les plus utilisés dans la littérature pour modéliser des prix de l'immobilier, à savoir la GWR et le krigeage, on peut voir que les performances sont comparables, la différence des erreurs sur la période de validation étant non significative (Figure 6.30). (Les erreurs des modèles de GWR et krigeage sont celles calculées aux chapitres 4 et 5 respectivement.)

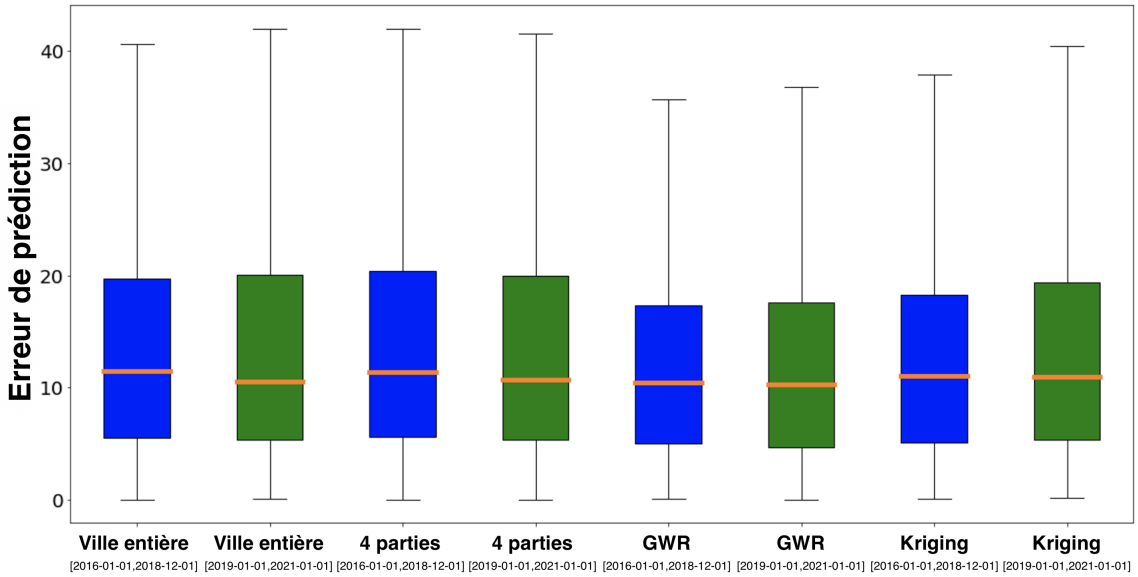


FIGURE 6.31 – Boites à moustache des erreurs de prédiction pour la ville de Paris

.

En regardant les distributions des erreurs sur la période T_{val} de la figure 6.31, on peut voir dans un premier temps qu'il n'y a aucune différence en termes de performance entre un apprentissage des paramètres global (boîte à moustache «Ville entière») et plus local (boîte à moustache «4 parties»). En effet, la variation entre les prix produits par la première optimisation et ceux produits par la dernière est extrêmement faible et donc va avoir un impact quasi nul sur les backtests. Cela est possiblement dû au fait que la ville est trop petite pour pouvoir espérer un gain en appliquant de l'apprentissage local. Aussi, il y a des mois où aucune transaction ne se passe dans une cellule et donc l'apprentissage peut être biaisé. Il faut noter que le coût de passage d'un apprentissage global à un apprentissage local est très faible pour une petite ville.

Si l'on compare maintenant notre nouveau modèle (boite à moustache «Ville entière») avec les modèles de la littérature (GWR et krigeage), il n'y a pas de différence sur l'erreur médiane (Figure 6.31). Cependant, pour le pire des cas, notre nouveau modèle se trompe plus que le modèle de GWR (erreur maximum à 42% VS 37%). Cela est dû à la nature du modèle de GWR qui a tendance à lisser les prix.

6.5 Discussion

La modélisation d'une ville sous forme de réseau de neurones apprenant son marché immobilier semble prometteuse. En effet, les résultats obtenus sur nos deux cas d'étude, Paris et Les Lilas, sont encourageants car les performances sont déjà presque identiques à celles des modèles précédents alors même que nous n'utilisons qu'une première approche naïve : la structure du réseau est simplement la structure géographique des localisations. De plus, le nombre de paramètres du modèle étant faible (paramètre spatial et paramètre temporel) cela permet une facilité d'interprétation des résultats obtenus, ainsi qu'une plus grande rapidité d'optimisation. Autre avantage lié au modèle, sa rapidité d'exécution, puisque, après l'initialisation, les mises à jours s'effectuent simplement à partir des transactions du mois directement courant — alors que les autres modèles repartent chaque mois de 4 années de données antérieures. Pour illustrer la comparaison, les erreurs des trois modèles développés dans cette thèse sont affichées sur la figure 6.32.

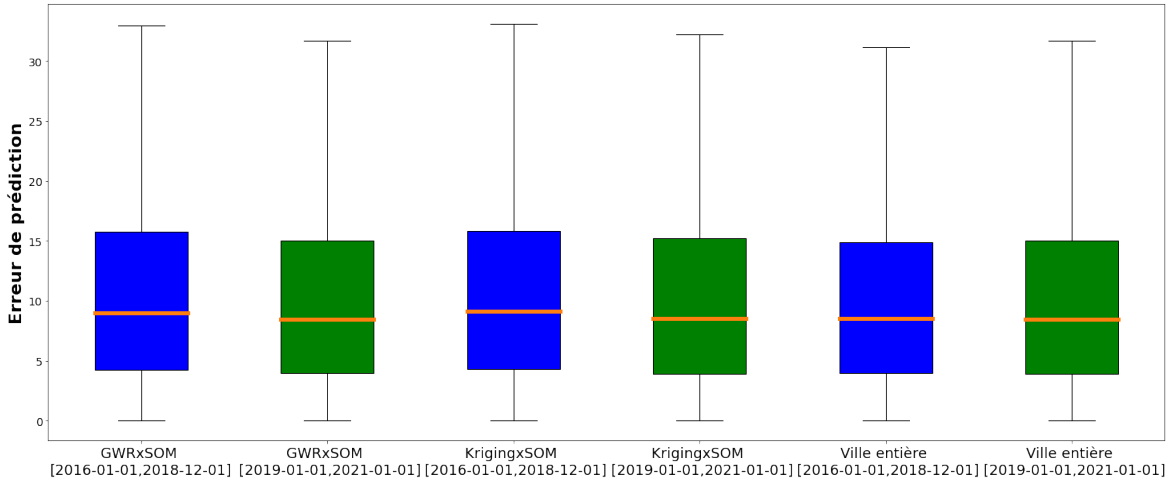


FIGURE 6.32 – Distribution des erreurs de prédiction à Paris

Les modèles de diffusion spatiale (GWR, krigeage) qu'ils soient augmentés par l'information de clustering ou non, repartent chaque mois de 4 années de données antérieures, comme nous le disions plus haut, ce qui « coûte cher » lors de l'estimation. Comme pour tout modèle spatial, un arbitrage est fait entre la contrainte de mémoire et celle du calcul des distances à chaque itération en fonction du modèle et des capacités de mémoire/calcul de l'utilisateur. Dans le cadre de cette thèse, on a ainsi dû calculer une matrice des distances pour toutes les parcelles d'Ile-de-France

pour un voisinage allant jusqu'à 1 500 mètres. Cela représente plus de 100 voisins par parcelle pour un total de 3,5 millions de parcelles. Les calculs de distance ont été faits en **SQL** et ont nécessité 12 heures de calcul et un espace de stockage de plusieurs centaines de gigaoctets. C'est un cas extrême où toutes les distances sont calculées pour un rayon de voisinage très élevé, pour toutes les villes d'Ile-de-France. Cette matrice de distance est utilisée pour la GWR et pour la construction du graphe de voisinage du modèle du chapitre 6. Les calculs de distances entre observations sont recalculés à chaque application du krigeage, car le décalage spatial sur les observations (pour pouvoir inverser la matrice des corrélations spatiales) ne nous permet pas d'utiliser la matrice directement. Le temps d'exécution de chaque modèle à été relevé pour une application à Paris au 1er janvier 2021. On prend comme exemple le cas de Paris, car il comporte une contrainte réelle à laquelle font face à la fois le milieu privé des entreprises et les laboratoires de recherche à savoir manipuler un volume de données massif. Dans l'ordre de temps d'exécution décroissant (nous n'incluons pas ici le temps d'exécution de SOM) :

- GWR : 1 heure 44 minutes. La matrice de distance est calculée post calcul des estimations et l'estimation pour une ville est parallélisée sur 4 coeurs, ce qui rend le calcul beaucoup plus rapide. Nous pourrions envisager l'option de parallélisation pour GWRxSOM, mais elle n'a pas été explorée ici par manque de temps.
- GWRxSOM : 4 heures. La matrice de distance est calculée post calcul des estimations, mais le nombre de localisations à estimer et le nombre d'observations à prendre en compte pour chaque localisation est grand.
- Krigeage : 30 minutes. L'utilisation des matrices creuses, des KDTree et de la fonction **krige** du package **gstat** sur le logiciel **R** rendent les calculs beaucoup plus rapides.
- KrigeagexSOM : 7 heures. Ce modèle n'utilise pas de KDTree ni de matrices creuses par manque de temps d'adaptation, mais théoriquement ces techniques pourraient être utilisées dans le cadre du krigeage augmenté. Le produit de matrice pour chaque localisation à estimer est fait en **python** et n'est pas optimisé.
- Réseau de neurones : 7 minutes pour la création du graphe et 27 secondes pour une itération. Le graphe est créé à une date d'initialisation à partir de la matrice de distance présentée plus haut et traduit la structure géographique

de la ville. Il est donc calculé une seule fois pour une ville donnée et reste inchangé. Le stockage de l'état du réseau à chaque itération est possible et se fait quasi instantanément. Le temps de calcul dépend du volume de donnée par itération, mais il ne dépasse jamais 40 secondes pour Paris.

La méthode basée sur le réseau de neurones présente l'avantage d'être bien plus rapide que les autres méthodes en plus d'être mathématiquement beaucoup plus simple. De plus, elle s'abstrait d'un biais dont les autres modèles peuvent être victimes en incorporant directement la dimension temporelle dans la définition du modèle. En effet, les modèles de GWR ou krigeage nécessitent une actualisation des prix des transactions à la date du modèle à partir des indices de prix de l'immobilier. Ces indices eux-mêmes sont calculés à partir de modèles hédoniques et sont difficiles à évaluer. L'erreur de prédiction de ces deux modèles peut donc parfois être liée à l'indice utilisé.

Étant donné la simplicité et la rapidité du modèle par réseau de neurones, il est donc aisé d'effectuer des optimisations locales, voire ultra-locales pour analyser la formation des quartiers : quelles tailles ont-ils en fonction de la localisation dans la ville ? Il est aussi intéressant d'étudier le couple (σ,α) pour l'ensemble des partitions de la ville. Nous n'avons pas développé d'outil pour cette analyse par manque de temps.

Beaucoup de pistes d'amélioration existent pour ce modèle, la première portant directement sur la définition du graphe. En effet, pour ne pas se limiter au cas où le graphe reflète une structure purement géographique de la ville, il faudrait apprendre la structure du graphe en fonction des dynamiques passées des prix qui peuvent nous renseigner sur la diffusion à travers la ville. La détection des frontières de quartiers se ferait aussi par apprentissage de la diffusion des prix des données passées et les arêtes du graphe seraient pondérées par cette nouvelle information. On pourrait en outre envisager l'étude sur une ville abstraite, où l'on ferait des changements de structure afin de mesurer la sensibilité de la diffusion du prix.
Même sans mettre en œuvre un tel apprentissage de structure, on pourra déjà améliorer sans doute les résultats qui ont été obtenus avec une définition des parcelles immédiatement voisines comme étant celles dans un voisinage de 25 mètres : c'est un choix de paramètre un peu arbitraire qui mériterait aussi d'être calibré par *grid*

search par exemple, ou alors on pourrait aussi simplement s'abstraire d'une définition a priori du voisinage en pondérant les arrêtes du graphe par la distance géographique uniquement — cependant cela changerait la notion de diffusion, la propagation de l'information de deux immeubles séparés par un boulevard serait plus lente que celle de deux immeubles mitoyens.

Un des enjeux qui n'a pas été exploré ici est l'adaptabilité du modèle, en particulier, dans un autre contexte de données. Les données sur lesquelles le modèle a été appliqué sont des informations sur les transactions remontées par les agences partenaires de MeilleursAgents. On a donc à disposition des bases de données complètes en termes de variables descriptives des biens, mais présentant un biais de représentativité géographique. Selon les localisations, il serait parfois plus intéressant d'utiliser la base publique des Demande de Valeur Foncière (DVF) censée être exhaustive, mais celles-ci présentent beaucoup moins de caractéristiques descriptives utiles pour nous (nombre de pièces et surface seulement).

D'autres questions sur l'adaptabilité d'un modèle dans des zones où le marché est très peu dynamique par exemple, et sur les conditions de mise en production sont détaillées dans le chapitre suivant.

Chapitre 7

Une nouvelle méthode de création d'indices de prix de l'immobilier chez Meilleurs Agents

Meilleurs Agents (MA) produit trois niveaux d'informations sur le marché immobilier français, à savoir des indices d'évolution des prix pour une zone géographique, des cartes de prix qui reflètent l'estimation du prix m^2 d'un bien standard et un outil d'estimation pour n'importe quel bien. Le projet présenté dans ce chapitre a nécessité plus d'un an de recherche et développement et est directement relié au premier niveau d'information : celui des indices de prix de l'immobilier. Il a été élaboré avec Carmélo Michiche, doctorant en économie à l'Université de Cergy-Pontoise et Data Scientist chez MA et développé par des ingénieures de l'équipe Data-Science, Laura Manzke et Tania Situm.

Les indices étant utilisés pour prendre en compte l'évolution d'un phénomène dans le temps, cela suppose que l'on puisse observer le phénomène en question sur un échantillon suffisamment représentatif tout au long d'une période donnée. Or en ce qui concerne le domaine de l'immobilier, les données à disposition sont difficilement accessibles, rare et de nature hétérogène. Si la construction d'un indice ne pose pas de problèmes pour un marché dynamique où il y a suffisamment de transactions pour que la qualité de l'indice soit bonne (à condition d'utiliser une bonne méthode), comment faire pour refléter les dynamiques de marchés en ayant peu d'information ?

Nous souhaitons construire des indices partout en France au plus proche des dynamiques des marchés locaux, c'est-à-dire créer des indices les plus granulaires possible avec une temporalité faible afin de pouvoir capter les évolutions qui correspondent le plus à la réalité. Il est donc nécessaire que chaque indice soit calculé sur une zone géographique contenant un volume de données suffisant. Une manière de pallier le manque de données est d'augmenter la fenêtre temporelle pour laquelle les observations sont agrégées, seulement si celle-ci est trop grande, 6 mois par exemple, l'indice ainsi crée risque de manquer les variations à court terme. Un autre moyen est d'agrandir l'entité géographique considérée en passant des transactions d'une ville à celle d'un département par exemple. Or même dans ce cas il y a des limites, comme pour le département de la Creuse où il n'y a pas assez de transactions. Nous cherchons donc une maille géographique permettant de couvrir un maximum de territoire avec suffisamment de données, pour cela un clustering des villes ayant des dynamiques de prix similaires serait pertinent.

On retrouve ici un besoin similaire au problème de diffusion spatiale d'une information de prix que nous avons rencontré dans les chapitres précédents de cette thèse. Comme nous l'avons vu, la création d'une mesure de similarité socio-économique entre des zones géographiques via l'algorithme SOM se révèle pertinent pour diriger ce processus de diffusion, dans le but de faire ressortir la structure spatiale des prix dans une ville. L'objectif ici est différent puisqu'il s'agit de réunir des entités géographiques similaires, au sens de leur évolution, mais l'utilisation de la même technique pour quantifier ce degré de similarité reste pertinent. Par ailleurs, au même titre que pour la production de carte, cette mesure de similarité socio-économique ne peut être utilisée seule mais doit être croisée avec la distance physique, dans la mesure où c'est en premier lieu selon cette dimension que s'organise le marché immobilier résidentiel. En l'occurrence, elle est introduite en limitant les agrégations au niveau du département ou de la région pour ne pas mélanger des marchés trop différents.

Nous présentons dans un premier temps la manière dont nous appliquons l'algorithme SOM pour former des clusters de quartiers/villes homogènes. L'enjeu principal est de produire les clusters les plus petits possible, c'est-à-dire de minimiser le nombre d'entités géographiques par clusters tout en respectant le bon volume de données pour pouvoir construire un indice convenable. Ainsi, il faut aussi définir le seuil de volume de données minimum. Nous détaillons ensuite les étapes de création

d'un indice de prix de l'immobilier pour chaque cluster à travers l'utilisation du modèle hédonique. Cette nouvelle méthode permet d'avoir des indices de prix de l'immobilier pour n'importe quelle ville en France, ils sont directement accessibles sur le site Meilleurs Agents.

7.1 Utilisation de SOM pour le regroupement d'entités géographiques

7.1.1 Les données

Les sources de données utilisées pour ce projet sont de trois natures différentes. La première est la base des biens vendus (BV) dont les transactions sont remontées par les agences partenaire de MA partout en France. Elle présente un biais de représentativité, n'est pas exhaustive, mais est à jour les transactions étant remplies de manière journalière. Les données provenant des bases publiques de Demandes de Valeurs Foncière (DVF) publiées par la Direction Générale des Finances Publiques (DGFIP) est quant à elle exhaustive (sauf pour l'Alsace-Moselle ou aucune donnée n'est disponible), mais fournie très peu de variables caractéristiques des biens (prix de la transaction et sa date de vente, surface et nombre de pièces uniquement). De plus, elle est mise à jour avec un décalage temporel de plus de 6 mois entre la date de vente et la réception de la donnée. La troisième source est la base BIEN de Paris Notaire Service (appelé *notaire* dans la suite), elle concerne les transactions se situant en Île-de-France uniquement. Cette base aussi est exhaustive, l'information relative aux caractéristiques des biens est remplie par les notaires directement et il y a peu de valeurs manquantes, mais elle aussi présente un décalage temporel de 6 mois. Il est précisé ici que les sources ne sont pas mélangées, mais sont utilisées de manière indépendante pour la construction d'indices propre à chaque source.

7.1.2 Articulation des bases entre elles

Les deux bases *notaire* et DVF étant censées contenir l'entièreté des transactions passées, on pourrait penser qu'il est préférable d'utiliser la base DVF uniquement puisqu'elle est disponible partout en France. Seulement, le nombre de variables caractéristiques des biens sur cette base étant limité, nous préférons travailler avec la

base *notaire* en Île-de-France (IDF) qui contient plus de variables caractéristiques et avec la base DVF partout ailleurs.

Afin d'avoir un indice couvrant toute une période et à jour, l'indice créé à partir de bases exhaustives (*notaire* ou DVF) est prolongé avec l'indice créé à partir des bases internes (BV) pour les 6 derniers mois (ou plus pour la base DVF) où cette donnée n'est pas disponible. Il y a toujours plus de transactions issues de la base *notaire* ou DVF que de transactions BV quelque soit la zone. Le clustering fait en utilisant les volumes $notaire/DVF$ est donc une subdivision du clustering BV.

Les clusters sont ainsi calculés pour deux granularités géographiques : au quartier et à la ville, pour les appartements et maisons séparément. Il pourra donc y avoir : 2 (source *notaire* ou $DVF + BV$) $\times$ 2 (niveaux géographiques) $\times$ 2 (type de bien) = 8 indices différents par entité géographique. Un exemple de clustering de quartier pour deux sources (BV et *notaire*) est montré dans le tableau 7.1.

TABLE 7.1 – Exemple d'un clustering BV et *notaire* pour six quartiers différents, pour le type appartement. Dans cet exemple, les quartiers A,B et C se retrouvent dans le même cluster BV (cluster 1). Les quartiers A et B restent ensemble dans le cluster *notaire* (cluster 4) mais le quartier C se retrouve seul (cluster 5).

type	quartier	cluster BV	cluster *notaire*
appartement	A	1	4
appartement	B	1	4
appartement	C	1	5
appartement	D	2	6
appartement	E	2	7
appartement	F	3	8

7.1.3 Description de l'algorithme de clusterisation

L'objectif est de construire les clusters les plus petits possible en nombre d'entités sous la contrainte d'une taille suffisante en termes de nombre de transactions pour y créer un indice. Un seuil minimum de transactions est déterminé sur la base de la qualité d'un indice en fonction du volume de données à disposition. La qualité étant mesurée à partir de la volatilité de l'indice. Un exemple de la définition du

seuil minimum pour les quartiers à Paris est montré dans la Figure 7.1.

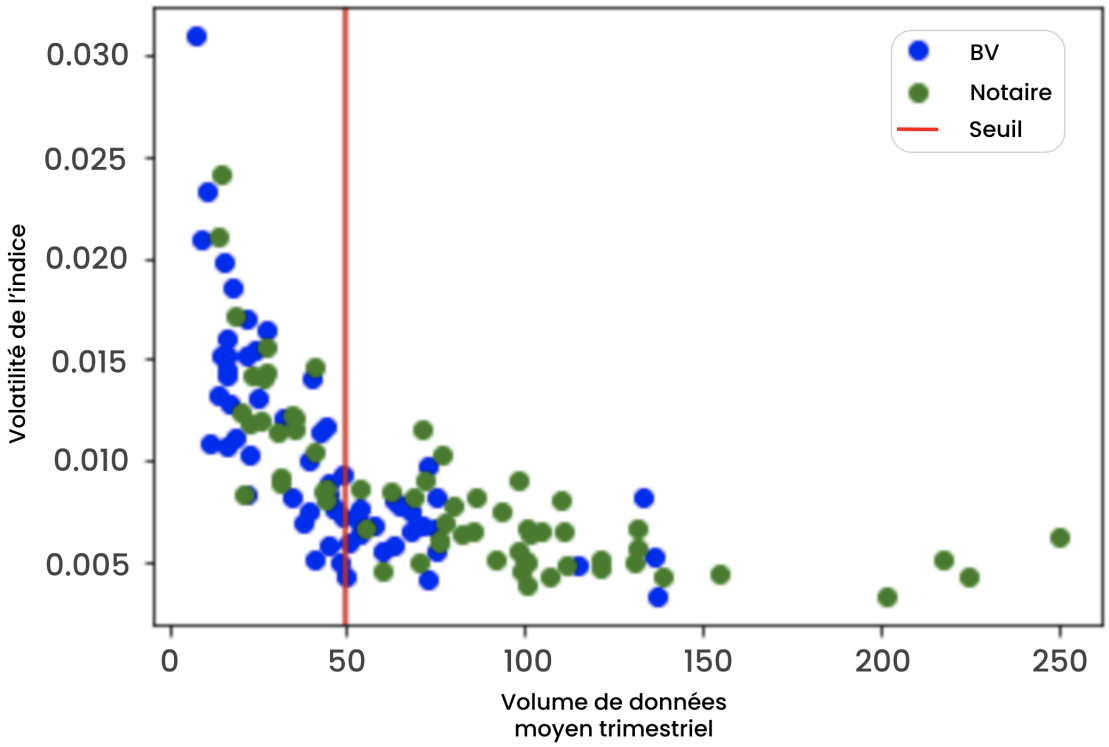


FIGURE 7.1 – Volatilité en fonction du volume moyen de données par trimestre pour les sources *BV* et *notaire*, pour chaque quartier à Paris. Chaque point représente un quartier, nous remarquons que les indices les moins volatiles sont ceux ayant été construits avec un minimum de 50 de données par trimestre.

Le seuil minimum est le même selon le type (appartement ou maison) et l'entité géographique (quartier ou ville) mais diffère selon la source :

TABLE 7.2 – Seuil minimum pour la construction d'un indice cluster en fonction des sources

source	seuil minimum
BV	50 données en moyenne par trimestre
notaire	50 données en moyenne par trimestre
DVF	100 données en moyenne par trimestre

On applique dans un premier temps l'algorithme SOM au niveau des quartiers/villes à partir de variables socio-économiques provenant des bases de l'INSEE, de la même manière que pour les chapitres précédents. Ces variables sont décrites dans le tableau 7.3. Nous exécutons 100 itérations de l'algorithme SOM et nous

choisissons l'itération qui minimise le critère d'inertie intra-classe. À l'issue de cette première étape de clustering, il est possible que certains clusters n'aient pas un volume suffisant pour pouvoir y construire un indice. Le volume de données d'un cluster correspond à la somme des volumes calculés par entités (quartier/ville) appartenant à ce cluster.

Entité géographique	Label	Description
Ville	med_rev	revenu médian de la ville
	prop_cadre	proportion de cadres dans la ville
	prop_ouvrier	proportion d'ouvriers dans la ville
	prop_chomeurs	proportion de chômeurs dans la ville
	prop_retraite	proportion de retraités dans la ville
	prop_maison	proportion de maisons dans la ville
	densite_logement	densité de résidences principales construites dans la ville (en km^2)
	prop_logav19	proportion de résidences principales construites avant les années 1900
	prop_log4670	proportion de résidences principales construites entre 1946 et 1970
	prop_log7190	proportion de résidences principales construites entre 1971 et 1990
Quartiers	dec1	1er décile de la distribution du revenu dans la ville
	dec1	1er décile de la distribution du revenu dans la ville
	med	revenu médian
	dec9	9ème décile de la distribution du revenu dans la ville
	patr	part du revenu provenant du patrimoine
	minsoc	part du revenu provenant des minimas sociaux

TABLE 7.3 – Description des variables utilisées pour appliquer l'algorithme SOM au niveau du quartier et de la ville. Il faut noter que pour le cas des quartiers nous utilisons des variables disponibles à l'IRIS, puis nous agrégeons ces données au quartier en faisant une moyenne pondérée par le parc.

On procède dans un second temps à une étape d'agrégation des clusters issus de SOM via une CAH.

Un dendrogramme est construit à l'aide des distances SOM entre clusters. Le dernier cluster de ce dendrogramme (constitué de l'ensemble des clusters SOM) est coupé si les clusters issus de cette séparation dépassent les seuils, c'est-à-dire s'il y a suffisamment de données pour avoir des clusters plus fins. Les clusters issus de l'étape précédente sont eux aussi sous-découpés si les sous-clusters dépassent les seuils, cette dernière étape est répétée de manière itérative tant que des clusters dépassent les seuils.

Rappelons ici que les bases *notaire* et *DVF* ayant un décalage temporel et étant plus complètes, leurs clusters sont le résultat d'une subdivision des clusters *BV*. Afin de créer ces nouveaux clusters *notaire* et *DVF*, le dendogramme continue d'être coupé, seulement le critère d'arrêt est celui du volume de données *notaire* ou *DVF* (par opposition au volume BV).

Le contexte de données étant différent en Ile-de-France et hors de l'Ile-de-France, les règles de clustering sont différentes selon la localisation dans laquelle on se trouve.

7.1.4 Cas de l'Ile-de-France

Pour le cas de l'Île de France, le clustering se limite aux villes appartenant au même département afin de garder une notion de proximité géographique et ainsi ne pas mélanger des marchés trop différents. Pour chaque département nous appliquons un clustering sur les villes pour la source BV puis *notaire*, pour les appartements et maisons. Chaque ville v du département d appartient à un unique cluster selon la source et le type. Le département est séparé en n_d clusters avec $C = (C_1, ..., C_{n_d})$ et il est défini par :

$$d = \bigcup_{k=1}^{n_d} \bigcup_{v \in C_k} v \tag{7.1}$$

La clusterisation au niveau des quartiers concerne uniquement la ville de Paris, mais la création de clusters quartiers pourrait être étendue à d'autres villes d'Ile-de-France. Chaque quartier b appartient à un unique cluster et on a la partition de Paris en n_c clusters avec $C = (C_1, ..., C_{n_c})$. En adaptant 7.1, Paris est défini est définit comme l'union des clusters qui la compose.

7.1.5 Cas hors de l'Île-de-France

Il est nécessaire d'adopter une autre logique pour les cas hors de l'Île-de-France car il y existe des départements pour lesquels le volume cumulé des transactions n'atteint pas le seuil minimum pour construire un indice. On effectue dans un premier temps un clustering sur les villes appartenant aux Aires Urbaines suffisamment grandes. D'après l'INSEE [31], une Aire Urbaine (AU) est un ensemble de communes, d'un seul tenant et sans enclave, constitué par un pôle urbain (unité urbaine) de plus de 10 000 emplois, et par des communes rurales ou unités urbaines (couronne périurbaine) dont au moins 40 % de la population résidente ayant un emploi travaille dans le pôle ou dans des communes attirées par celui-ci.

Les départements ne dépassant pas le seuil du volume de données minimum sont réunis et aucun clustering n'est appliqué.

Il existe ensuite un clustering au niveau des quartiers pour chacune des dix plus grandes villes de France, pour les appartements uniquement. Ici encore, le choix a été fait de créer des indices quartiers pour les dix plus grandes villes de France pour des questions métiers, mais cette granularité de clustering peut-être étendue à d'autres villes.

À l'issue de ces étapes, la France entière est recouverte par des clusters à la ville, pour le cas des appartements et des maisons. La Figure 7.2 réunis les clusters *DVF* et *notaire* pour le cas des appartements, sur la France entière.

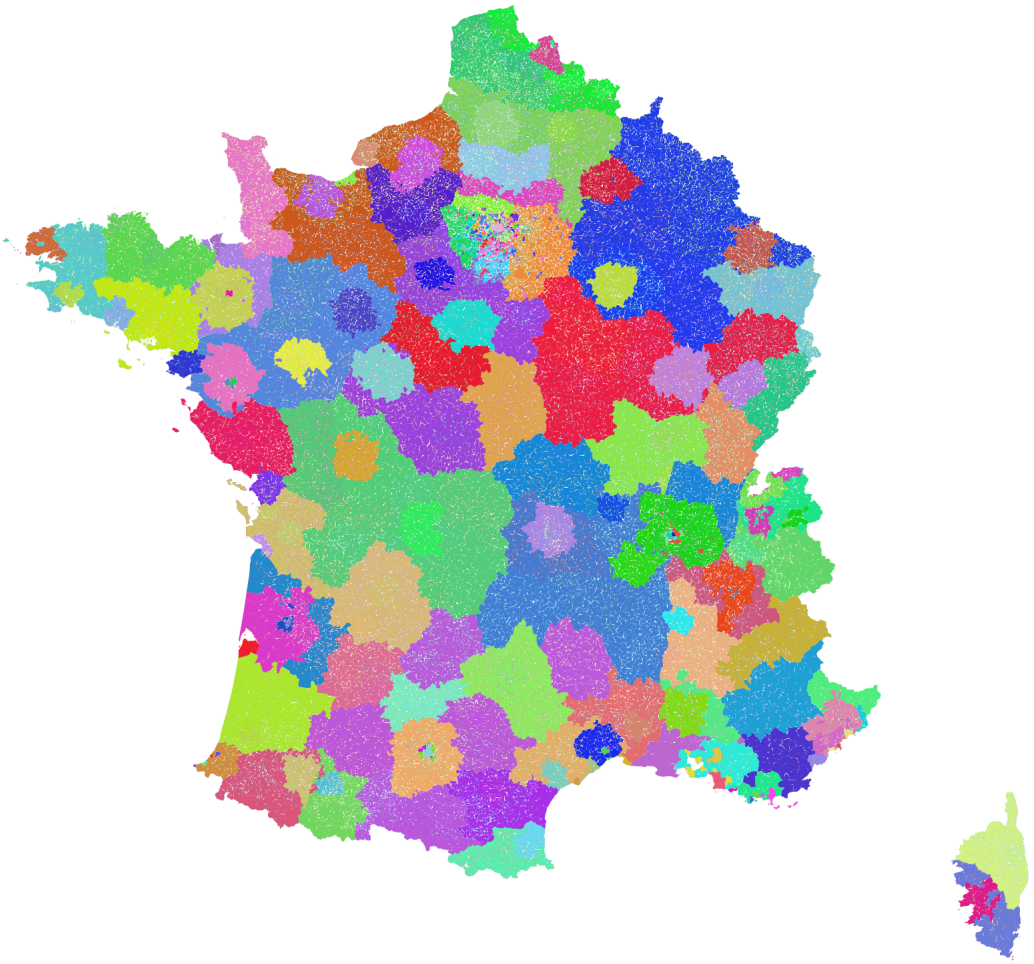


FIGURE 7.2 – Clusters des villes de France pour la source *DVF* et *notaire* et pour le type appartement.

7.2 Création d'indices

La section qui suit décrit succinctement la méthodologie hédonique de création des indices. Le lecteur s'intéressant plus particulièrement à cette méthodologie et à la justification des choix faits et la mesure de leurs impacts est renvoyé vers *A methodology for local housing price index in France* de Micciche et al. [10].

7.2.1 Prétraitement des données

Des filtres sont d'abord appliqués sur les caractéristiques telles que le prix, le nombre de pièces, la surface pour éliminer les biens atypiques. Des quantiles sont

ensuite calculés dynamiquement chaque année afin d'avoir un ensemble de données homogène, par exemple on enlève 5% des biens les plus et moins chers par année. Les variables descriptives catégorielles sont ensuite transformées en des variables binaires pour chaque occurrence. Ces variables diffèrent selon la source (*BV*, *notaire*, *DVF*).

7.2.2 Régression hédonique

Nous choisissons d'appliquer une régression sur 36 mois avec un contrôle temporel trimestriel, il y a donc 12 périodes $T1, T2, ..., T12$. L'hypothèse hédonique stipule que l'apport d'une caractéristique sur le prix est constant, seulement elle cesse d'être valide sur le long terme : c'est pour quoi nous limitons la régression à 36 mois. L'ensemble X regroupe les variables propres aux caractéristiques du bien et les variables géographiques, P_i désigne le prix m^2 du bien i. La régression hédonique standard pour estimer le prix d'un bien est donnée par :

$$\log(P_i) = \beta_0 + \sum_{k=1}^{K} \beta_k X_i^k + \sum_{t=1}^{12} \gamma_t T_i^t \tag{7.2}$$

La période de référence est fixée à $T1$. Les valeurs V que composent l'indice pour chaque $t \in \{1, ..., 12\}$ sont définies par :

$$V_t = \begin{cases} V_{t-1} + (\gamma_t + \beta_0)/\beta_0 \times 100 & \text{si } t > 1 \\ 100 & \text{si } t = 1 \end{cases} \tag{7.3}$$

On a donc l'indice I trimestriel donné par :

$$I = [100, V_2, ..., V_{12}] \tag{7.4}$$

7.2.3 Mensualisation de l'indice trimestriel

En vue d'une mensualisation, nous créons trois régressions hédoniques trimestrielles telles que définies dans la section précédente décalées d'un mois de la date à laquelle on veut créer l'indice mensuel notées I_1, I_2 et I_3. Par exemple, si nous voulons créer un indice à la date de 2021-01-01 il faut créer :

- un indice trimestriel à 2021-01-01 : I_1
- un indice trimestriel à 2020-12-01 : I_2
- un indice trimestriel à 2020-11-01 : I_3

Pour un mois donné, on cherche son trimestre correspondant et on divise la variation de l'indice trimestriel par 3 pour avoir un indice mensuel. Les valeurs que compose l'indice mensualisé pour chaque mois $m \in \{1, ..., 36\}$ sont définies par :

$$V_m = \begin{cases} V_{m-1} + (I_t - I_{t-1})/3 & \text{si } m > 1 \\ 100 & \text{si } m = 1 \end{cases} \tag{7.5}$$

On effectue cette étape pour les trois indices trimestriels I_1, I_2, I_3. On note l'indice i mensualisé I_i^M et on pose :

$$I_i^M = [100, V_2, ...V_{36}] \ , \ i = 1, 2, 3 \tag{7.6}$$

Un bon proxy de l'indice mensuel final est de moyenner les trois indices mensualisés I_1^M, I_2^M, I_3^M. Pour cela, on calcule la moyenne des variations des indices pour chaque mois noté δ_m. Par exemple, la moyenne des variations au point 2020-11-01 c'est-à-dire au mois $m = 34$ est donnée par :

$$\delta_{34} = (\delta_{36}(I_3^M) + \delta_{35}(I_2^M) + \delta_{34}(I_1^M))/3 \tag{7.7}$$

où $\delta_m(I_i^M) = (I_{i,m}^M - I_{i,m-1}^M)/I_{i,m-1}^M$ avec $I_{i,m}^M$ la valeur au mois m de l'indice mensualisé I_i^M.

Les valeurs que compose l'indice final pour chaque mois $m \in \{1, ..., 36\}$ sont définies par :

$$V_m^F = \begin{cases} V_{m-1}^F + \delta_m & \text{si } m > 1 \\ 100 & \text{si } m = 1 \end{cases} \tag{7.8}$$

Et l'indice final I^F est donné par :

$$I^F = [100, V_2^F, ..., V_{36}^F] \tag{7.9}$$

Les indices étant crées sur des clusters de villes (ou de quartier) il faut recomposer

l'indice du département (ou de la ville) à partir des indices de ses clusters sous-jacent. En reprenant l'exemple de section 7.1.3 (équation 7.1) l'indice I_d du département d est défini comme étant la somme pondérée du parc (noté p) des n_d indices clusters villes :

$$I_d = \frac{\sum_{k=1}^{K} I_k \times p_k}{\sum_{k=1}^{K} p_k} \tag{7.10}$$

où I_k et p_k sont l'indice et le parc du cluster C_k, le parc étant le nombre de logements, il faut différencier le cas où considère des appartements du cas des maisons.

7.2.4 Hybridation d'indices provenant de sources différentes

Les bases *notaire* et *DVF* ayant un retard temporel de 6 mois, il est nécessaire de trouver un moyen pour produire des indices à jour chaque mois. Une première méthode serait de prédire l'évolution de l'indice des prix sur les 6 derniers mois en se basant sur les tendances passées, car les indices sont autocorrélés, cependant elle ne permet pas de suivre ni la saisonnalité ni les ruptures de tendance.
Nous faisons le choix ici de simplement ajouter l'indice construit à partir des bases BV à l'indice provenant des bases dites exhaustives, car cela permet d'avoir un indice réactif aux changements de tendances.

Soit m le mois courant et $m-6$ six mois auparavant, I^E, I^B les indices finaux tel que décrit dans 7.9 pour les bases exhaustives (*notaire* ou DVF) et pour la base BV respectivement. L'indice de sources hybride I^H est défini par :

$$I^H = I^E + [\delta^B_{m-6}, .., \delta^B_m] \tag{7.11}$$

où $\delta^B_i = (I^B_i - I^B_{i-1})/I^B_{i-1}$.

7.3 Résultats

On applique l'algorithme présenté en section 7.1 sur toutes les villes de France et on obtient 6 types de clusters différents (clustering maison/appartements sources

BV, *notaire* et *DVF*). La Figure 7.3 réunis les clusters *DVF* et *notaire* pour le cas des maisons, sur la France entière. Les villes des grandes Aires Urbaines se retrouvent souvent dans le même et unique cluster sauf pour le cas des grandes villes (Marseille et Lyon par exemple) et les villes des départements peu dynamiques se retrouvent rassemblées ensemble (Figure 7.3).

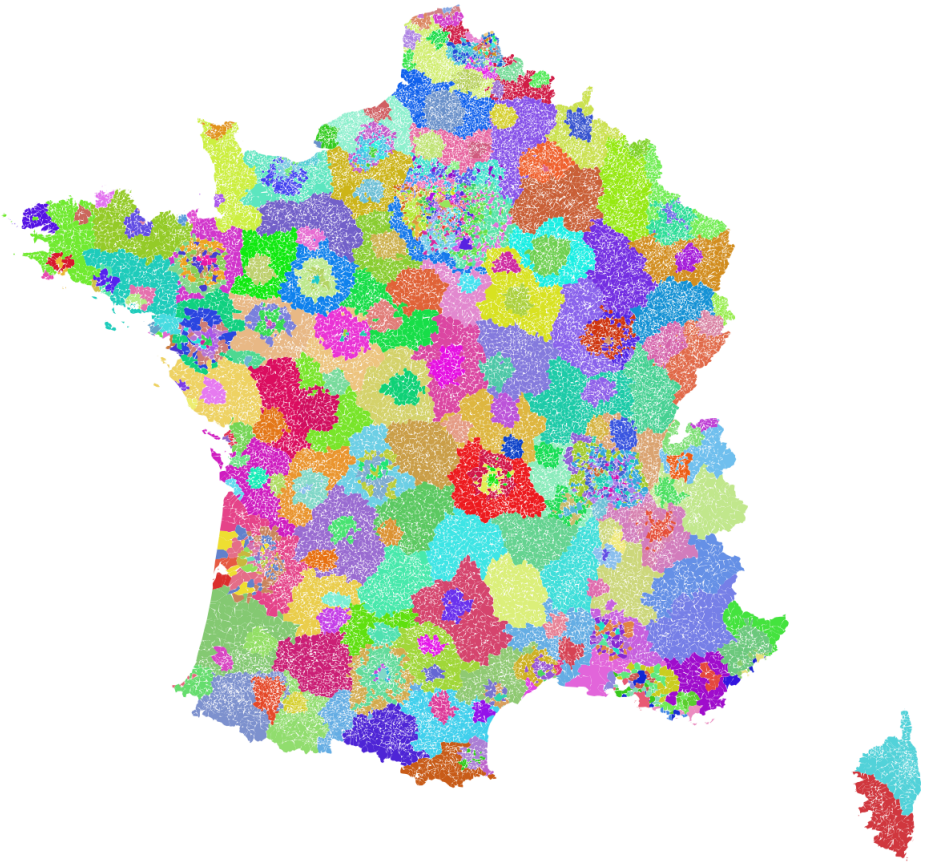


FIGURE 7.3 – Clusters des villes de France pour la source *DVF* et *notaire* et pour le type maison.

Les clusters formés à partir des quartiers à Paris sont présentés dans la Figure 7.4 pour la source *notaire*. La contrainte d'avoir des clusters les plus petits possible en ayant un nombre de transactions suffisant est ici respectée (Figure 7.4). En effet, les petits quartiers du centre de la ville sont réunis et les quartiers plus grands sont seuls dans leur cluster (certains quartiers du 15ème ou du 11ème arrondissement par exemple).

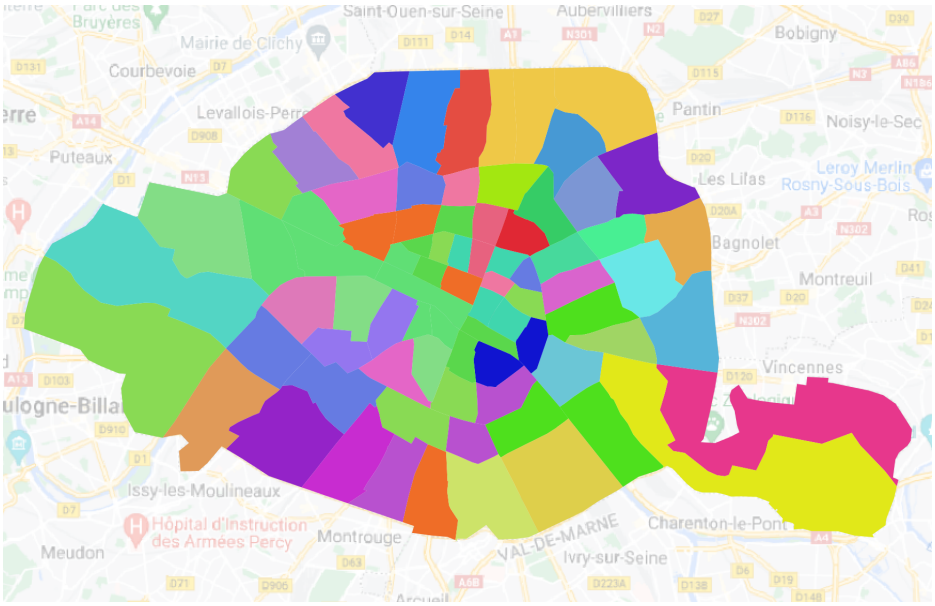


FIGURE 7.4 – Clusters des quartiers à Paris pour la source *notaire*.

Le clustering appliqué aux Aires Urbaines de grandes villes produit un découpage généralement assez clair comme c'est le cas pour Bordeaux par exemple (Figure 7.5). On distingue trois partitions nettes avec une partie plus rurale (en bleu), une première couronne (en orange) et Bordeaux avec deux villes limitrophes (Talence et Le Bouscat).

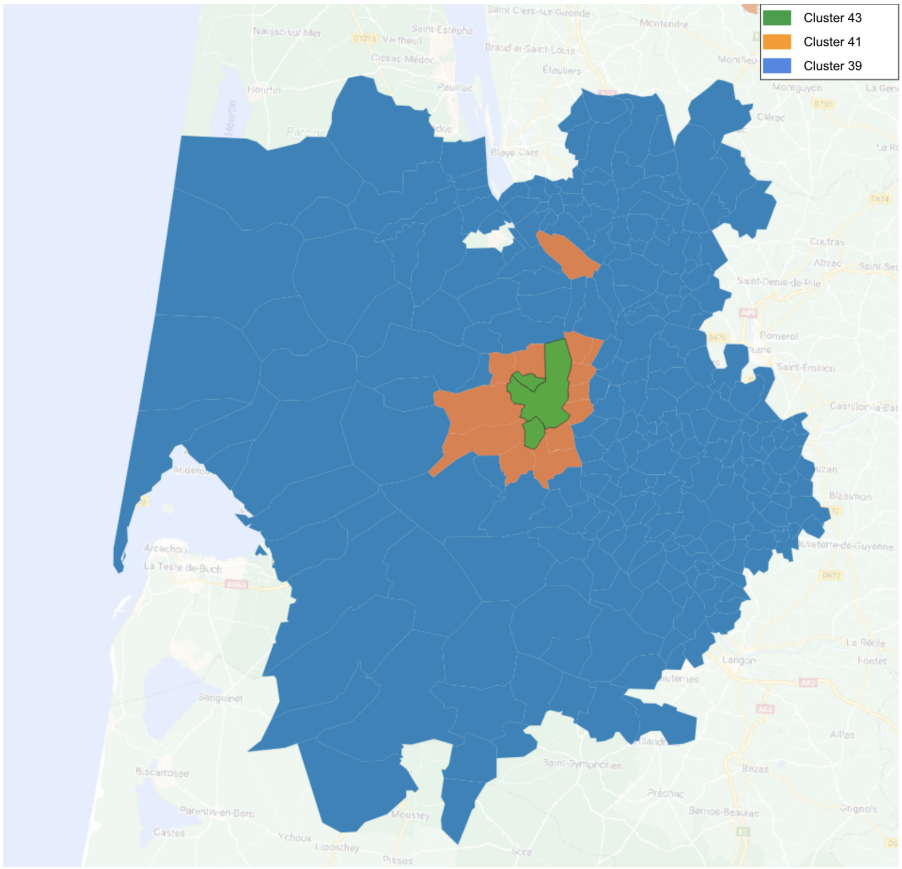
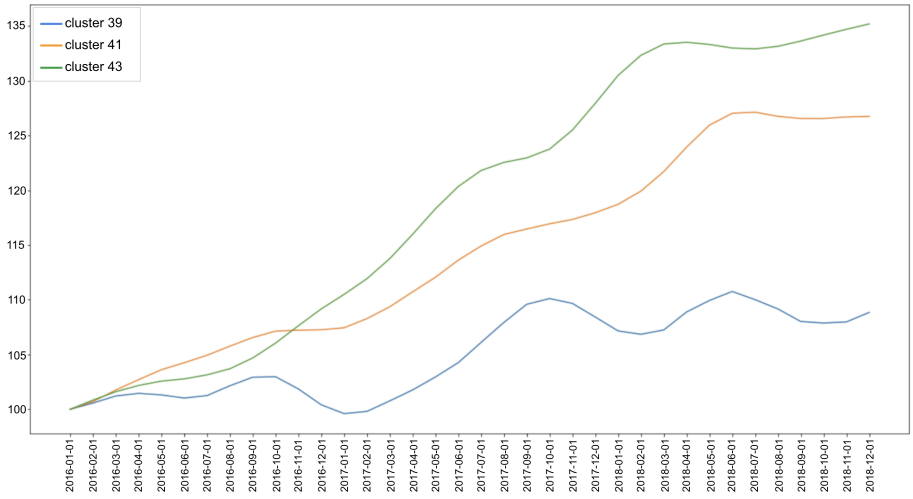
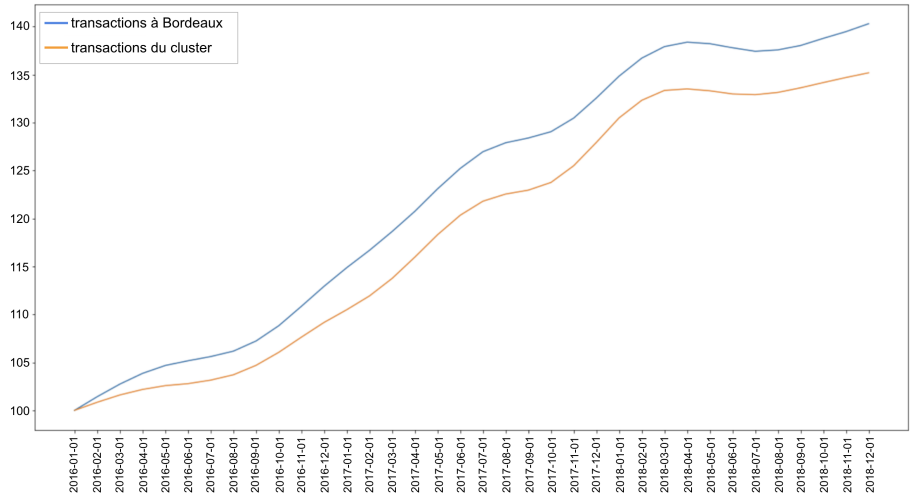


FIGURE 7.5 – Clusterisation de l'AU de Bordeaux pour la source *DVF* et le type appartement.

Cette séparation du territoire de manière nette se traduit aussi sur les indices, allant du plus dynamique pour le cluster des villes de Bordeaux au moins dynamique pour la partie rurale (Figure 7.6(a)). En comparant l'indice produit avec les transactions se trouvant à Bordeaux unique et celui construit à partir de celles appartenant au cluster réunissant Bordeaux, Talence et Le Bouscat, on remarque que ces deux dernières villes n'ont pas beaucoup d'influence sur la dynamique globale du cluster cas les indices sont très similaires (Figure 7.6(b)).



(a) Indices des prix de l'immobilier pour les clusters appartenant à l'AU de Bordeaux, pour la source *DVF* et le type appartement au 1er décembre 2018.



(b) Indices des prix de l'immobilier formés à partir des transactions d'appartements issues de la source *DVF*, pour des transactions à Bordeaux uniquement (indice bleu) et pour celles appartenant au cluster 43 (indice orange).

FIGURE 7.6 – Indices des prix de l'immobilier à Bordeaux et son Aire Urbaine au 1er décembre 2018.

En dehors de l'Île-de-France, le regroupement des quartiers est effectué seulement pour les 10 plus grandes villes de France. Néanmoins, le choix qui a été fait d'appliquer un clustering sur toute une Aire-Urbaine peut amener des villes comme Montpellier (Figure 7.7) à se retrouver avec d'autres villes au sein du même cluster. La clusterisation des quartiers se fait donc pour l'ensemble des quartiers dont les

villes sont dans le même cluster. Il y a donc autant d'indices que de clusters et ils ont des dynamiques différentes (Figure 7.8(b)). L'indice de Montpellier qui en découle prendra donc les dynamiques des quartiers voisins, mais sera limité, car l'indice des clusters quartier est pondéré par la représentativité de ceux-ci, en termes de nombre de logements, lorsque l'on souhaite reconstruire l'indice de la ville (Figure 7.8(c)).

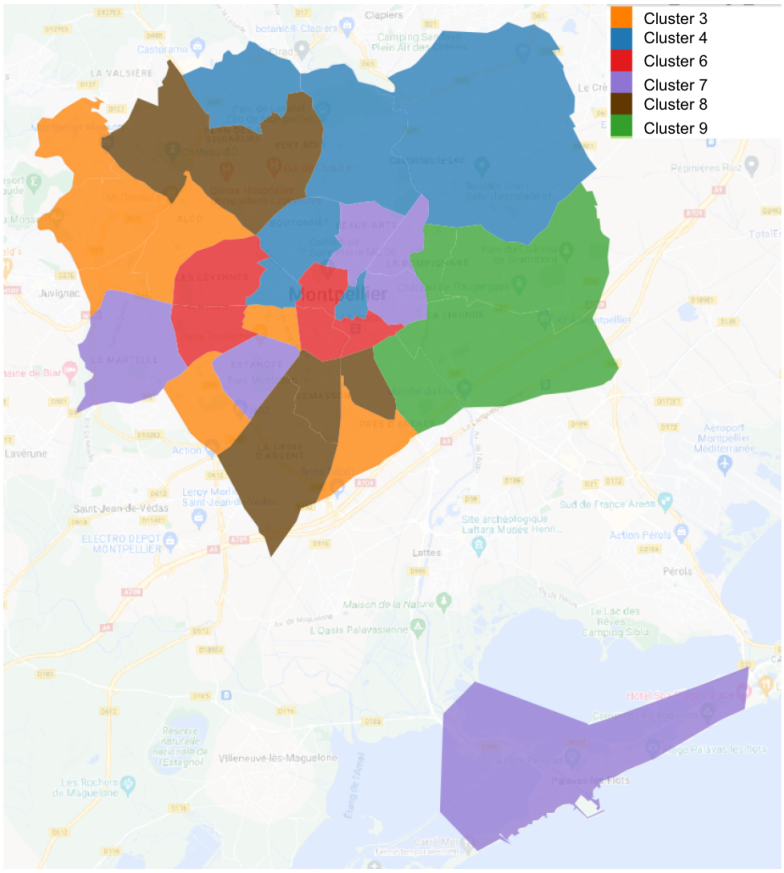
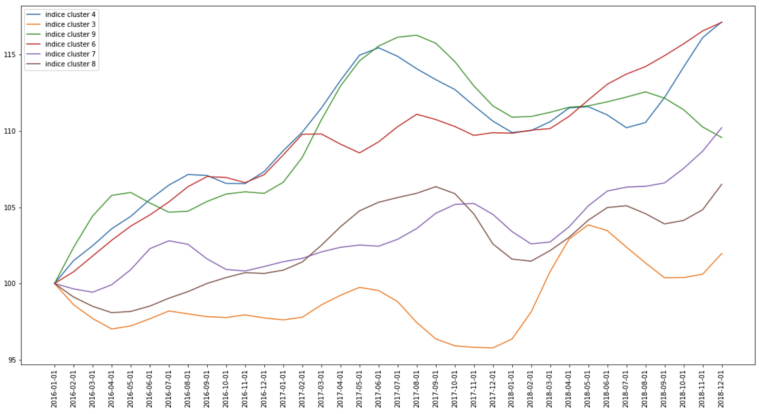


FIGURE 7.7 – Clustering des quartiers pour Montpellier et les villes appartenant à son cluster, soit Palavas-les-Flots et Castelnau-le-Lez, pour la source *DVF* et pour le type appartement.



(a) Indices des clusters quartiers appartenant au cluster des 3 villes, pour la source *DVF* et pour le type appartement au 1er décembre 2018.

(b) Indices des prix de l'immobilier formés à partir des transactions d'appartements issues de la source *DVF*, pour des transactions à Montpellier uniquement (indice bleu) et pour celles appartenant au cluster (indice orange).

FIGURE 7.8 – Indices des prix de l'immobilier à Montpellier et les villes de son cluster au 1er décembre 2018.

7.4 Discussion

En appliquant l'algorithme SOM pour former des clusters de villes et quartiers en France, puis en agrégeant ces derniers afin d'atteindre un volume de donnée minimum via une CAH, nous construisons des indices avec l'utilisation du modèle hédonique partout en France pour représenter l'évolution du marché immobilier des maisons et des appartements.

La mise en place d'une nouvelle logique de construction d'indice basée sur un clustering d'entités géographique permet d'avoir des clusters les plus petits possible et 1463 indices existent désormais. Ces indices respectent les contraintes initialement fixées, car ils sont les plus granulaires possible, avec une temporalité faible et sont construit avec un volume de donné suffisant. Ils reflètent ainsi les dynamiques des marchés locaux. Ces indices ont été mis en production en septembre 2022 et sont disponibles sur le site MeilleursAgents.

Tel qu'elle a été présentée, la méthode prend en compte la notion de proximité spatiale en tant que «filtre» et contraint d'adapter la granularité de ce filtre en fonction de la localisation : limite géographique du clustering au département pour l'Île-de-France, clustering sur des AU et filtre à la région hors de l'Île-de-France. Pour éviter de devoir adapter la méthode tout en conservant la notion de proximité spatiale, Micciche et al. développent une méthode de construction de clusters contigus géographiquement [10].

Le nombre d'indices produits ayant considérablement augmenté il faut adapter la maintenance en condition opérationnelle de ces nouveaux indices. Comment s'assurer chaque mois que la mise à jour des indices reflète la réalité du marché ? S'il était possible de passer en revue la majorité des indices, aujourd'hui ce n'est plus le cas et il faut avoir recours à des méthodes basées sur des règles et de l'apprentissage statistique afin d'automatiser ce processus.

Conclusion

Cette thèse est l'aboutissement de plusieurs années de travail sur les questions liées à la dynamique spatiale des prix et à la manière dont se diffuse l'information immobilière. Elle a été effectuée au sein de l'entreprise MeilleursAgents ce qui a permis d'avoir accès à des bases de données très conséquentes.

Trois nouveaux modèles ont été développés pour prédire des prix immobiliers, ils font l'objet d'un chapitre chacun dans ce manuscrit. Dans un premier temps, le croisement de la GWR et du krigeage avec l'algorithme SOM a visé à améliorer des modèles existants en passant par un raffinement sans en changer la structure fondamentale. Les résultats montrent que même si nos méthodes capturent une réalité qui est difficilement accessible autrement et fournissent des cartes de prix plus fines et plus complexes, elles ne produisent pas une réelle amélioration des prédictions. Une large marge d'amélioration demeure sans doute dans une utilisation de données socio-économiques plus récentes et plus précises mais nous avons finalement préféré explorer une toute autre piste : celle basée sur l'idée que l'information que nous cherchions à capturer à travers les données socio-économiques est en fait déjà contenue dans les prix, qui contiennent également bien d'autres informations.

Partant, le changement de paradigme apporté par le modèle du chapitre 6, qui consiste à considérer une ville comme un réseau de neurones, apporte une modélisation simplifiée par rapport aux modèles existants. Avec un modèle plus simple et plus léger, on arrive aux mêmes performances que les modèles utilisés dans la littérature ou que leurs versions « augmentées » que nous avons proposés, et ce déjà dans une version naïve du nouveau modèle et avec un temps de calcul beaucoup plus court (pas plus de quelques minutes *vs* plusieurs heures).

La modélisation d'une ville en tant que réseau de neurones qui apprend son propre marché de l'immobilier ouvre en outre beaucoup de pistes. L'amélioration immédiate qu'il serait intéressant de mener est celle du raffinement du modèle graphique par apprentissage de structure. De la même manière que pour les deux pre-

miers modèles, on aimerait intégrer une proximité socio-géographique sans avoir une connaissance particulière de la ville ou sans avoir à apporter de la donnée en plus. Une solution serait d'apprendre le contexte socio-géographique par la manière dont les prix se sont diffusés dans le passé et de pondérer les arrêtes du graphe en fonction de ce qui a été observé. Les arêtes ne seraient donc plus purement géographiques, mais traduiraient une proximité socio-économique en plus de la proximité spatiale. La question de l'évolution de la structure au cours du temps se posera : est-il nécessaire de la mettre à jour chaque mois, chaque année ? Probablement, il sera pertinent de suivre en parallèle l'évolution de la structure avec les mêmes techniques d'apprentissage que celles utilisées pour sa construction initiale, et éventuellement de la modifier continûment dans des villes très dynamiques.

La mise en production de ce modèle dans le cadre industriel pose également plusieurs questions et nécessite des tests supplémentaires. Notamment celle de l'adaptabilité du modèle pour un contexte différent, en particulier en zone très peu dense. Comment optimiser les paramètres pour un réseau où il peut n'y avoir aucune donnée en entrée pendant plusieurs itérations ? Dans ce cas, comment traduire la tendance des prix sans avoir de nouvelles observations ? Y a-t-il des cas où construire une carte des prix avec une granularité géographique fine est inutile ? Si oui, comment adapter le réseau à un autre niveau de localisation ? Autre question : l'étude a été appliquée pour estimer le prix des appartements à Paris et aux Lilas, néanmoins en France les maisons représentent 55.1% du parc immobilier (d'après l'INSEE), ce qui correspond à une part importante de l'audience du site MeilleursAgents. Comment donc adapter ce modèle pour l'estimation des maisons ? Une des pistes serait d'augmenter le rayon r qui détermine les parcelles voisines (égale à 25 mètres pour les appartements) pour l'adapter au cas des maisons où la taille du jardin/terrain impacterait beaucoup ce paramètre. Et puis, comment traiter le cas des maisons de ville ?

Bien sûr, pour le développement d'un modèle dans un contexte opérationnel, des arbitrages sont à faire entre prendre du temps pour mettre le modèle en production ou non, et le gain en termes d'erreur de prédiction est ce qui aide à prendre cette décision. Sur les villes étudiées, à savoir Paris et Les Lilas, il y a un gain de 1 point et 5 points respectivement sur l'erreur médiane par rapport au modèle aujourd'hui en production chez Meilleurs Agents. À Paris, l'abondance des données fait que l'erreur médiane est déjà faible (9% pour le modèle MA) alors qu'on peut

observer un vrai gain sur la ville des Lilas (15% pour le modèle MA). Nous avons vu par ailleurs au chapitre 7 que l'idée d'utiliser l'algorithme SOM (proposée au chapitre 4) pour réunir des entités géographiques similaires et produire une mesure de similarité socio-économique a pu être développée et mise en production dans le cadre de cette thèse, parallèlement à l'exploration du nouveau modèle du chapitre 6.

Terminons en évoquant une question non abordée dans cette thèse, celle du marché locatif. En 2019, la part des ménages locataires s'élève à 39.9% (INSEE). Quelles sont les interactions temporelles et spatiales entre niveaux de prix et loyers ? Ces interactions suffisent-elles à expliquer la dynamique des loyers ? Est-il réalisable d'appliquer ce modèle afin d'estimer les prix locatifs ? Il serait intéressant de comparer les paramètres de rayon de diffusion pour une même ville, par exemple à Paris où les prix des loyers sont beaucoup plus homogènes que les prix de vente.
Plus généralement, le nouveau paradigme proposé au chapitre 6 ouvre également des pistes très prometteuses pour l'étude des dynamiques socio-économiques urbaines : rien n'impose en effet de se limiter à l'étude de prix de l'immobilier, d'autres variables pourraient être considérées, et leurs dynamiques spatio-temporelles étudiées à travers ce modèle. Une étude plus générale de la formation et de l'évolution des quartiers, en particulier, pourrait être engagée.

Bibliographie

[1] Denis Allard. Statistiques spatiales : introductiona la géostatistique, 2012.

[2] Luc Anselin. *Spatial econometrics : methods and models*, volume 4. Springer Science & Business Media, 1988.

[3] Luc Anselin. Spatial externalities, spatial multipliers, and spatial econometrics. *International regional science review*, 26(2) :153–166, 2003.

[4] Daniel Arribas-Bel, Peter Nijkamp, and Henk Scholten. Multidimensional urban sprawl in europe : A self-organizing map approach. *Computers, Environment and Urban Systems*, 35(4) :263–275, 2011.

[5] Muhammad Ashraf, Jim C. Loftis, and K.G. Hubbard. Application of geostatistics to evaluate partial weather station networks. *Agricultural and Forest Meteorology*, 84(3) :255–271, 1997.

[6] Peter M Atkinson and Christopher D LLoyd. Non-stationary variogram models for geostatistical sampling optimisation : An empirical investigation using elevation data. *Computers & Geosciences*, 33, 2007.

[7] Sabyasachi Basu and Thomas G Thibodeau. Analysis of spatial autocorrelation in house prices. *The Journal of Real Estate Finance and Economics*, 17, 1998.

[8] Christopher Bitter, Gordon F Mulligan, and Sandy Dall'erba. Incorporating spatial variation in housing attribute prices : a comparison of geographically weighted regression and the spatial expansion method. *Journal of Geographical Systems*, 9, 2007.

[9] Chris Brunsdon, A. Stewart Fotheringham, and Martin E. Charlton. Geographically weighted regression : A method for exploring spatial nonstationarity. *Geographical Analysis*, 28(4) :281–298, 1996.

[10] Michel Baroni Carmélo Micciche, Pierre Vidal. A methodology for local housing price index in france. work in progress.

[11] Bradford Case, John Clapp, Robin Dubin, and Mauricio Rodriguez. Modeling spatial and temporal house price patterns : A comparison of four models. *The Journal of Real Estate Finance and Economics*, 29(2) :167–191, 2004.

[12] Martin E. Charlton, A. Stewart Fotheringham, and Chris Brunsdon. Geographically weighted regression. *Journal of the Royal Statistical Society Series D (The Statistician)*, pages 5–6, 2009.

[13] Wanfang Chen, Yuxiao Li, Brian J Reich, and Ying Sun. Deepkriging : Spatially dependent deep neural networks for spatial prediction. *arXiv preprint arXiv :2007.11972*, 2020.

[14] Jorge Chica-Olmo. Prediction of housing location price by a multivariate spatial method : cokriging. *Journal of Real Estate Research*, 29, 2007.

[15] Jorge Chica-Olmo and Rafael Cano-Guervos. Does my house have a premium or discount in relation to my neighbors ? a regression-kriging approach. *Socio-Economic Planning Sciences*, 72 :100914, 2020.

[16] Vincenzo Del Giudice, Pierfrancesco De Paola, Fabiana Forte, and Benedetto Manganelli. Real estate appraisals with bayesian approach and markov chain hybrid monte carlo method : an application to a central urban area of naples. *Sustainability*, 9(11) :2138, 2017.

[17] François Des Rosiers, Marius Thériault, and Paul-Y Villeneuve. Sorting out access and neighbourhood factors in hedonic price modelling. *Journal of Property Investment & Finance*, 2000.

[18] Allan Din, Martin Hoesli, and Andre Bender. Environmental variables and real estate prices. *Urban studies*, 38(11) :1989–2000, 2001.

[19] Michalis Doumpos, Dimitrios Papastamos, Dimitrios Andritsos, and Constantin Zopounidis. Developing automated valuation models for estimating property values : a comparison of global and locally weighted approaches. *Annals of Operations Research*, 306(1) :415–433, 2021.

[20] Robert Dubin and Chein-Hsing Sung. Specification of hedonic regressions : Non-nested tests on measures of neighborhood quality. *Journal of Urban Economics*, 27 :97–110, 1990.

[21] Robin Dubin. Predicting house prices using multiple listings data. *The Journal of Real Estate Finance and Economics*, 17 :35–59, 1998.

[22] Jean-Michel Floch. *Géostatistique*. INSEE, Paris, 2018.

[23] A Stewart Fotheringham, Chris Brunsdon, and Martin Charlton. *Geographically weighted regression : the analysis of spatially varying relationships*. John Wiley & Sons, Limited West Atrium, 2002.

[24] Laetitia Gauvin, Annick Vignes, and Jean-Pierre Nadal. Modeling urban housing market dynamics : can the socio-spatial segregation preserve some social diversity ? *Journal of Economic Dynamics and Control*, 37(7) :1300–1321, 2013.

[25] Matheron Georges. *Les variables régionalisées et leur estimation : une application de la théorie des fonctions aléatoires aux sciences de la nature / G. Matheron*. Masson, Paris, 1965.

[26] Kevin Gillen, Thomas Thibodeau, and Susan Wachter. Anisotropic autocorrelation in house prices. *The Journal of Real Estate Finance and Economics*, 23, 2001.

[27] Thibodeau Goodman. Housing market segmentation and hedonic prediction accuracy. *Journal of Housing Economics*, 12 :181–201, 2003.

[28] Natividad Guadalajara, Miguel Ángel López, Adina Iftimi, and Antonio Usai. Influence of the cadastral value of the urban land and neighborhood characteristics on the mean house mortgage appraisal. *Land*, 10(3) :250, 2021.

[29] Dean M Hanink, Robert G Cromley, and Avraham Y Ebenstein. Spatial variation in the determinants of house prices and apartment rents in china. *The Journal of Real Estate Finance and Economics*, 45(2) :347–363, 2012.

[30] Marco Helbich and Daniel A Griffith. Spatially varying coefficient models in real estate : Eigenvector spatial filtering and alternative approaches. *Computers, Environment and Urban Systems*, 57 :1–11, 2016.

[31] INSEE. Revenus, pauvreté et niveau de vie en 2015 - données carroyées, 2019.

[32] Teuvo Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43(1) :59–69, 1982.

[33] Teuvo Kohonen. *Self-Organizing Maps*. Springer Series in Information Sciences. Springer Berlin Heidelberg, 2012.

[34] Andy L Krause and Christopher Bitter. Spatial econometrics, land values and sustainability : Trends in real estate valuation research. *Cities*, 29 :S19–S25, 2012.

[35] Georges Matheron. Principles of geostatistics. *Economic geology*, 58(8) :1246–1266, 1963.

[36] William J McCluskey, Michael McCord, PT Davis, Martin Haran, and David McIlhatton. Prediction accuracy in mass appraisal : a comparison of modern approaches. *Journal of Property Research*, 30(4) :239–265, 2013.

[37] Richard Meese and Nancy E Wallace. The construction of residential housing price indices : A comparison of repeat-sales, hedonic-regression and hybrid approaches. *The Journal of Real Estate Finance and Economics*, 14(1-2) :51–73, 1997.

[38] R Kelley Pace, Ronald Barry, and Clemon F Sirmans. Spatial statistics and real estate. *The Journal of Real Estate Finance and Economics*, 17(1) :5–13, 1998.

[39] Marco Pangallo, Jean-Pierre Nadal, and Annick Vignes. Residential income segregation : A behavioral model of the housing market. *Journal of Economic Behavior & Organization*, 159 :15–35, 2019.

[40] Jorge Iván Pérez-Rave, Juan Carlos Correa-Morales, and Favián González-Echavarría. A machine learning approach to big data regression analysis of real estate prices for inferential and predictive purposes. *Journal of Property Research*, 36(1) :59–96, 2019.

[41] Julien Randon-Furling, Madalina Olteanu, and Antoine Lucquiaud. From urban segregation to spatial structure detection. *Environment and Planning B : Urban Analytics and City Science*, 2018.

[42] John H Relethford. Geostatistics and spatial analysis in biological anthropology. *American Journal of Physical Anthropology : The Official Publication of the American Association of Physical Anthropologists*, 136(1) :1–10, 2008.

[43] Sherwin Rosen. Hedonic prices and implicit markets : product differentiation in pure competition. *Journal of political economy*, 82 :34–55, 1974.

[44] Ay Se Can and Isaac Megbolugbe. Spatial dependence and house price index construction. *The Journal of Real Estate Finance and Economics*, 14 :203–222, 1997.

[45] Piyawan Srikhum. *Statistiques spatiales et étude immobilière*. PhD thesis, Université Paris Dauphine-Paris IX, 2012.

[46] Dieudonné Tchuente and Serge Nyawa. Real estate price estimation in french cities using geocoding and machine learning. *Annals of Operations Research*, 308(1) :571–608, 2022.

[47] Nicolas Thouvenin. *La formation des prix des logements anciens : les apports de la théorie des prix hédoniques*. PhD thesis, Université Paris 10 Nanterre, 2005.

[48] Agostino Valier. Who performs better ? avms vs hedonic models. *Journal of property investment & finance*, 2020.

[49] Haizhen Wen, Yilan Jin, and Ling Zhang. Spatial heterogeneity in implicit housing prices : Evidence from hangzhou, china. *International Journal of Strategic Property Management*, 21(1) :15–28, 2017.

[50] Ann D. Witte, Howard J. Sumka, and Homer Erekson. An estimate of a structural hedonic price model of the housing market : An application of rosen's theory of implicit markets. *Econometrica*, 47(5) :1151–1173, 1979.

[51] Joseph Awoamim Yacim and Douw Gert Brand Boshoff. Impact of artificial neural networks training algorithms on accurate prediction of property values. *Journal of Real Estate Research*, 40(3) :375–418, 2018.