

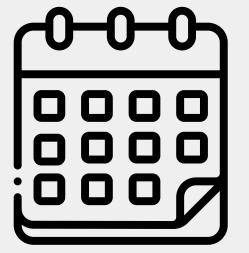
# ATELIER **DATASCIENCE**

**ABOUROUSSE - AMIOTTE - BERGER - BORDES**

# DATASET



- DVF (Demandes de valeurs foncières)
- Ventes immobilières
- Par année



- Uniquement 2023 pour notre analyse
- Comparaison avec les autres années

Fichiers (9)

Réutilisations (87)

Discussions (28)

## 5 FICHIERS PRINCIPAUX

### [Valeurs foncières 2023](#)

Mis à jour le 8 avril 2024 — txt (474.0Mo) — 36696 téléchargements

### [Valeurs foncières 2022](#)

Mis à jour le 8 avril 2024 — txt (596.5Mo) — 49700 téléchargements

### [Valeurs foncières 2021](#)

Mis à jour le 8 avril 2024 — txt (597.7Mo) — 20584 téléchargements

### [Valeurs foncières 2020](#)

Mis à jour le 8 avril 2024 — txt (449.2Mo) — 8316 téléchargements

### [Valeurs foncières 2019](#)

Mis à jour le 8 avril 2024 — txt (462.8Mo) — 14241 téléchargements

# ANALYSE ET NETTOYAGE DES DONNÉES

*Conversion des colonnes en float*

`['Valeur fonciere', 'Surface reelle bati', 'Surface terrain', 'Surface Carrez du 1er lot',  
 'Surface Carrez du 2eme lot', 'Surface Carrez du 3eme lot', 'Surface Carrez du 4eme lot',  
 'Surface Carrez du 5eme lot']`

*Ajustements et Filtrages*

*Remplacement des codes départementaux corses, suppression des lignes avec valeurs foncières manquantes et filtrages des données aberrantes*

*Remplacement des Nan*

*Remplacer les valeurs Nan dans le 'Nombre de pièces principales' en 0 et 'code type local' en 5.0*

# ANALYSE ET NETTOYAGE DES DONNÉES

*Suppression et création  
de colonnes*

*Colonnes de NaN / Type de lots  
Infos sur la voie / Info géographique*

*Somme des surfaces des  
lots*

*Surface du lot 1,2,3,... => Surface totale des lots*

*Regroupement des  
ventes*

*Ventes rassemblées avec le max des surfaces*

# RÉORGANISATION ET FEATURES

*Réorganisation des colonnes et suppression des lignes*

**Calcul du prix moyen du m<sup>2</sup> par département et communes**

**Réorganisation des colonnes pour une meilleure compréhension**

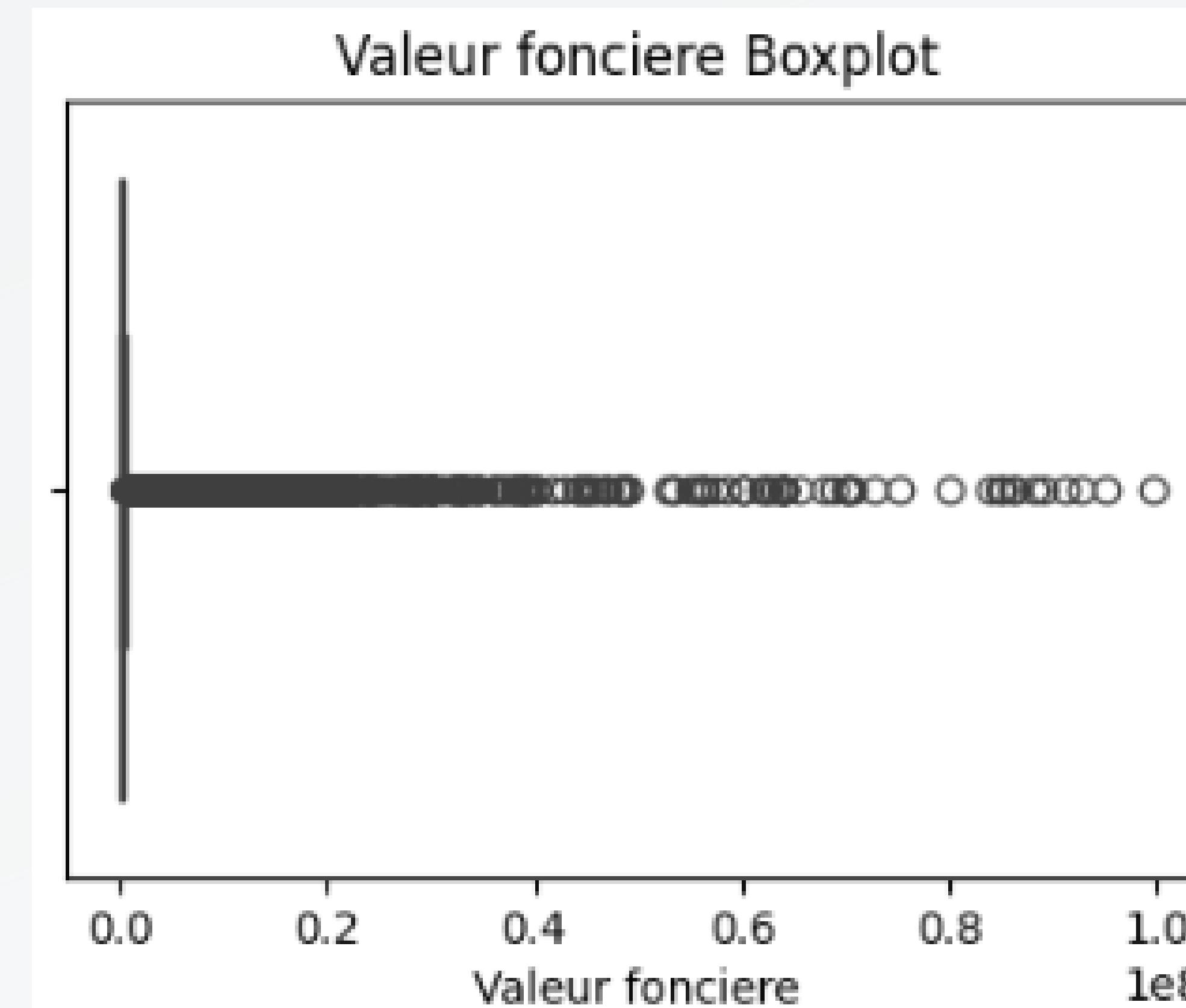
*Colonnes étudiées dans le dataframe*

**Date mutation** object → datetime  
**Nature mutation** str → int8  
**Valeur fonciere** float64  
**Code departement** int32  
**Code commune** int64  
**Code type local** float64  
**Nombre pieces principales** float64  
**Nombre de lots** int64  
**Surface totale des lots** float64  
**Surface reelle bati** float64  
**Surface terrain** float64

# ANALYSE DE DONNÉES

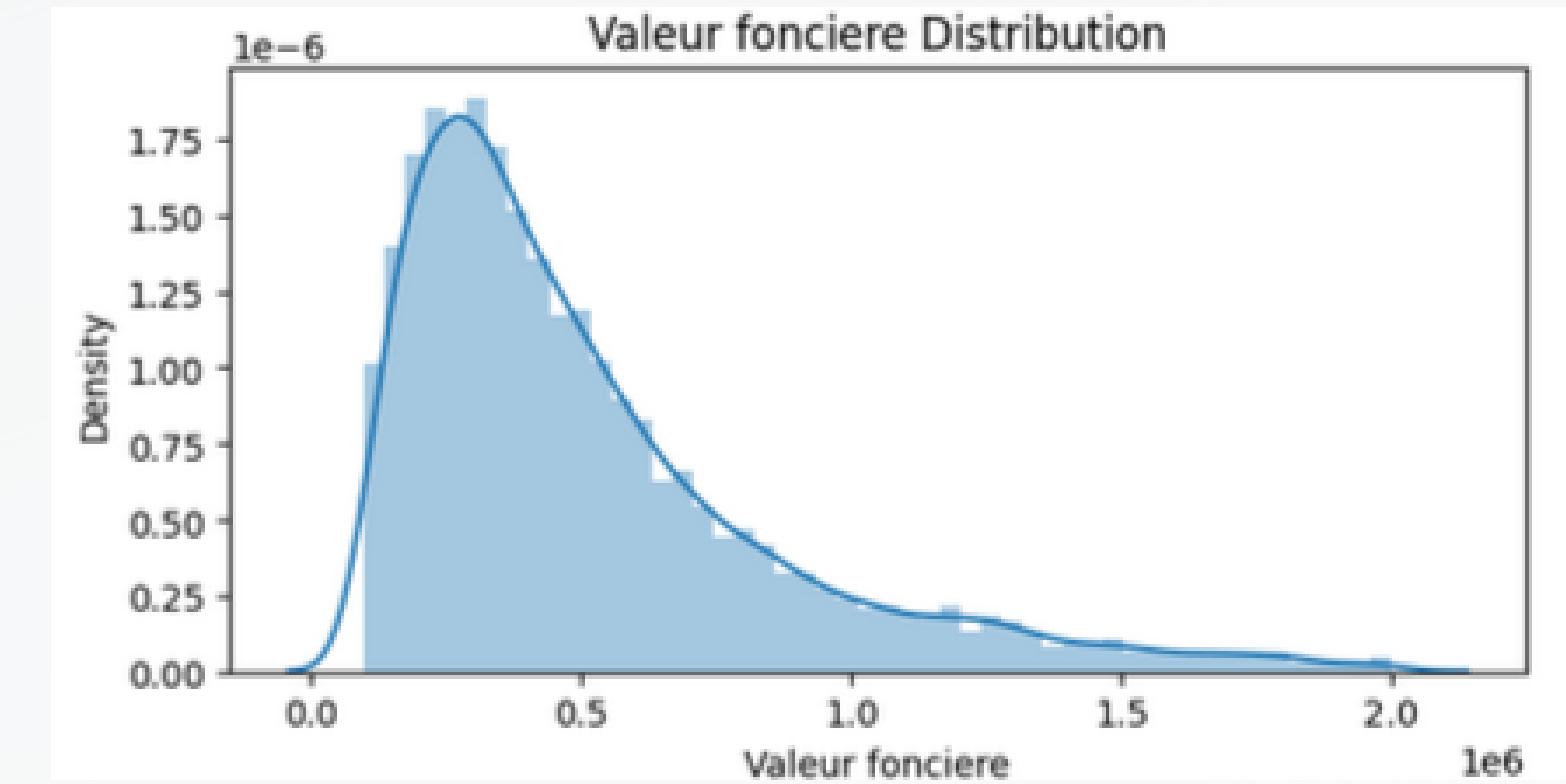
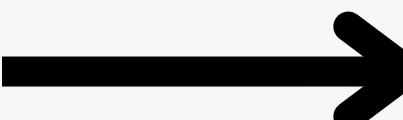
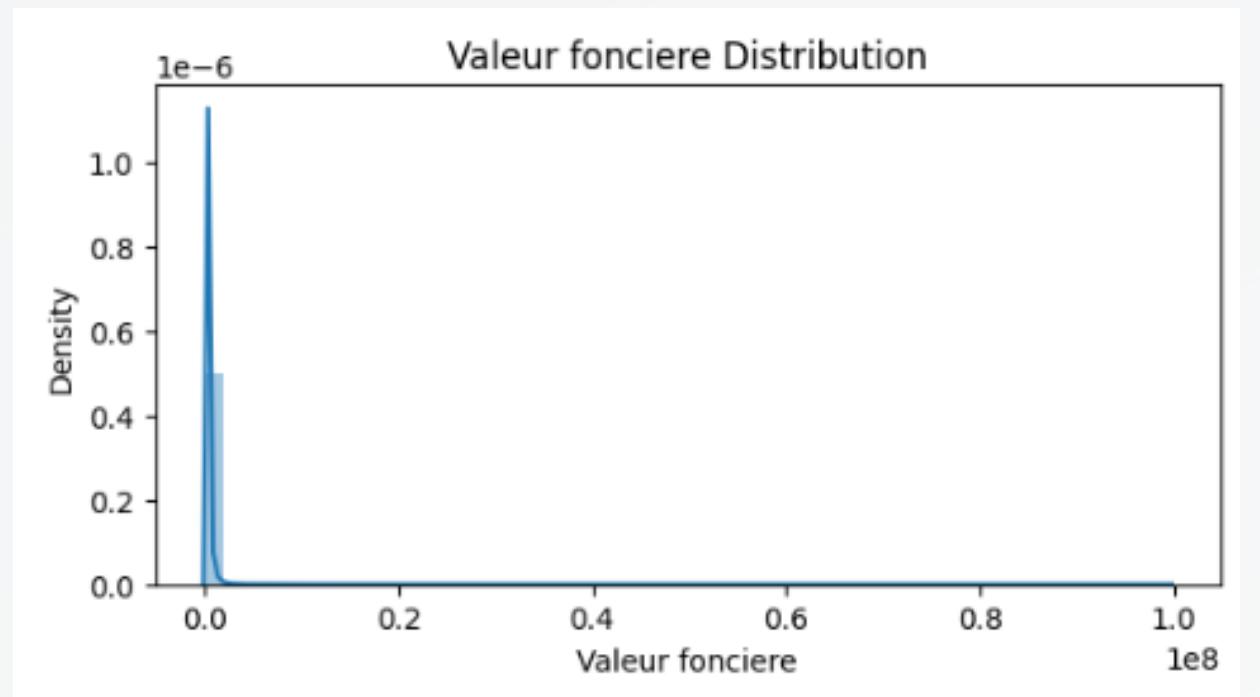
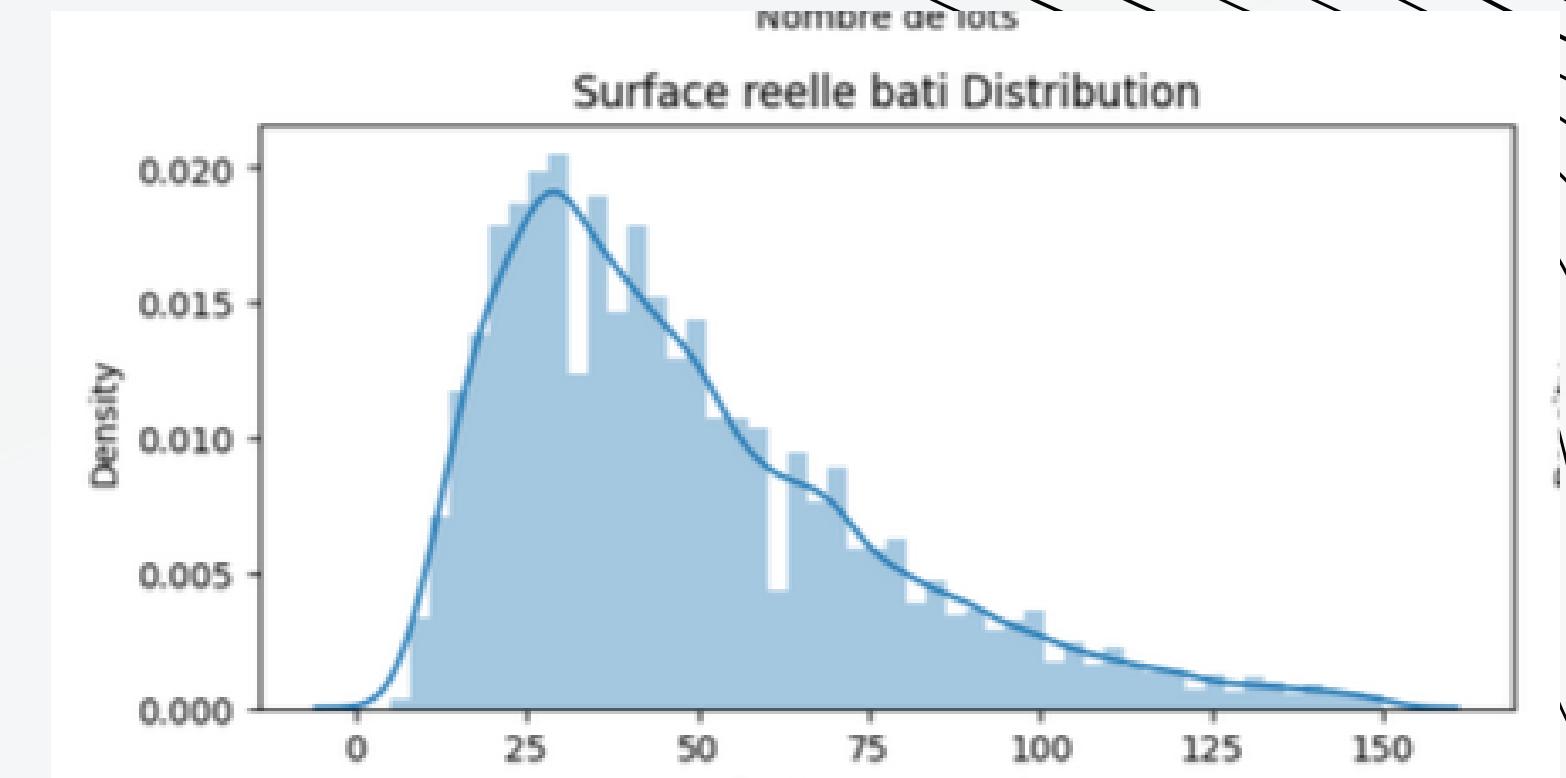
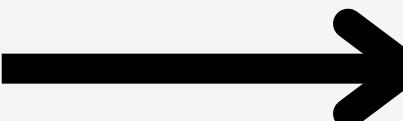
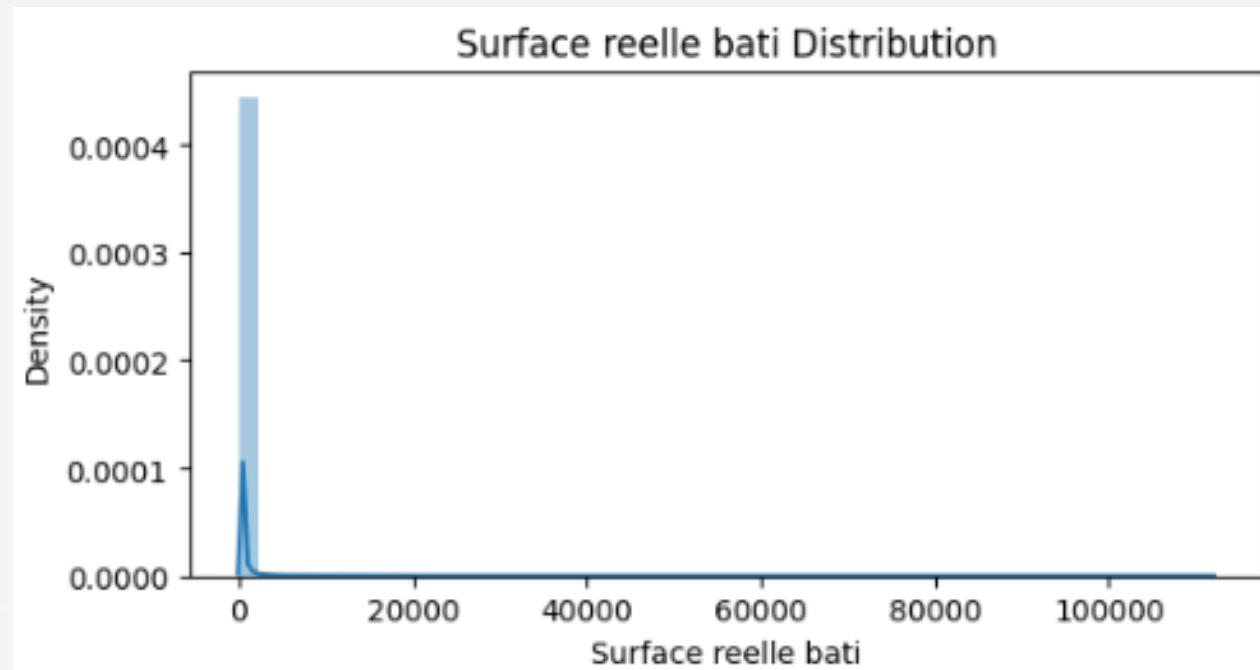


# VALEURS ABSURDES



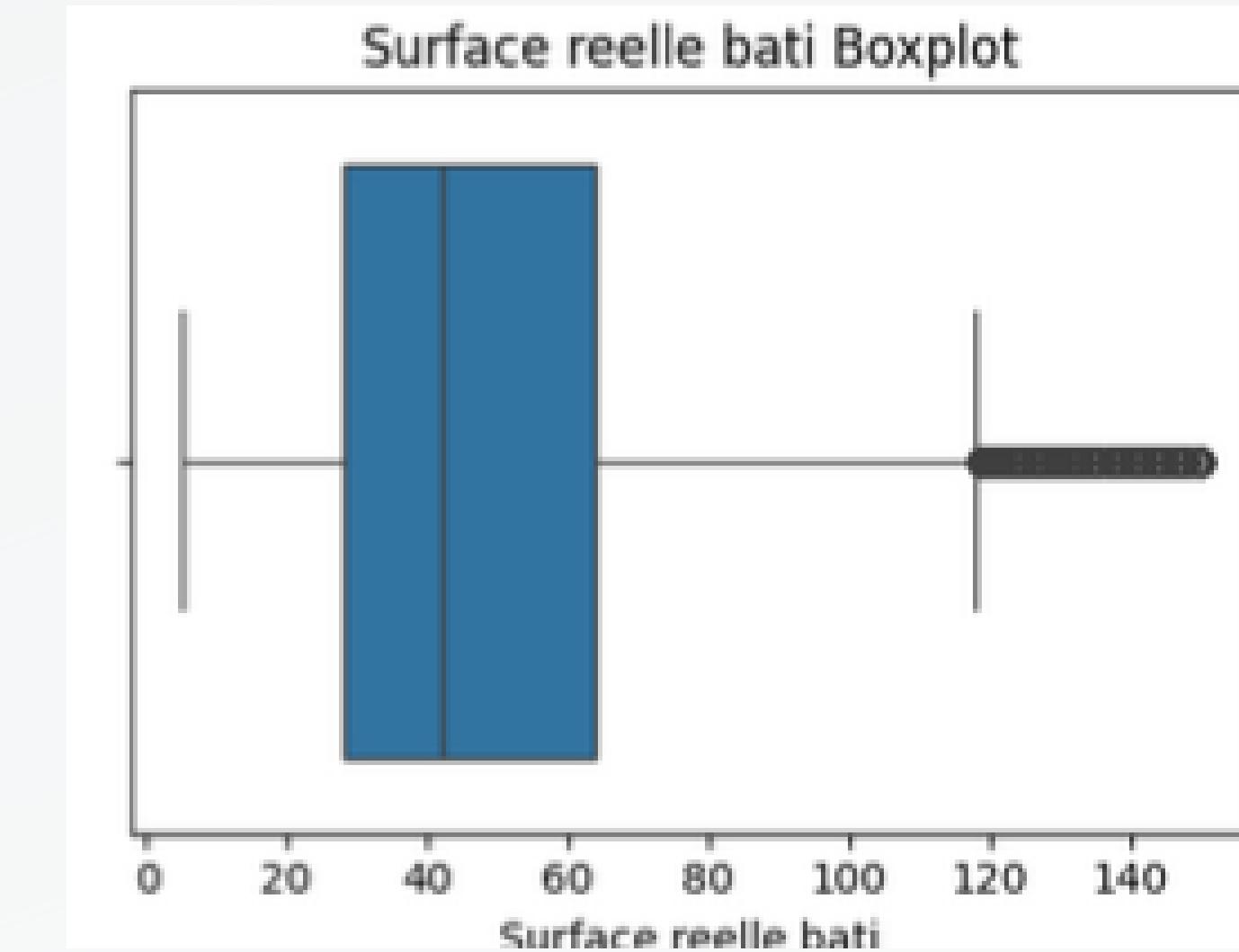
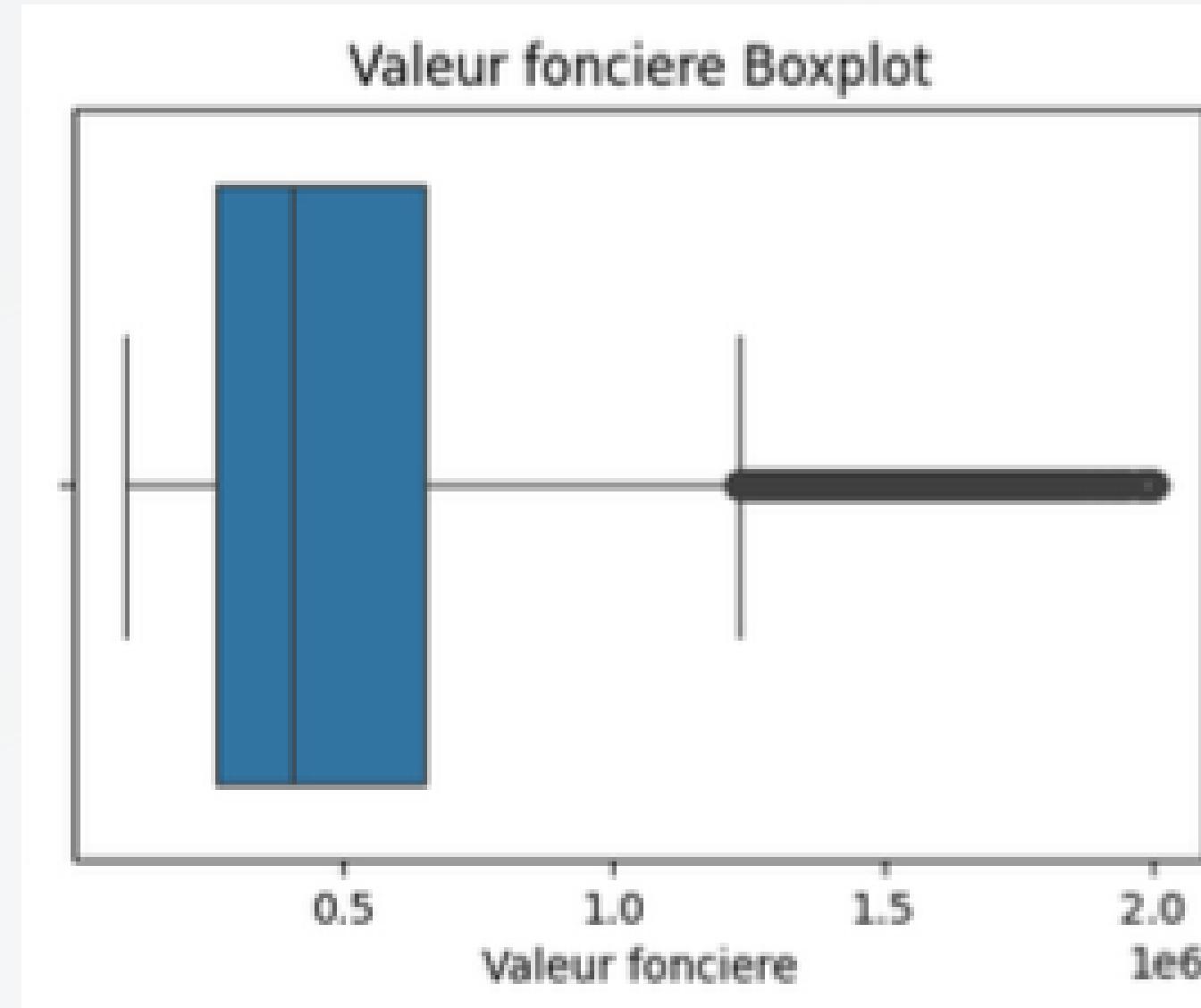
1er quartile 3eme quartile et moustache confondus

# VALEURS ABSURDES

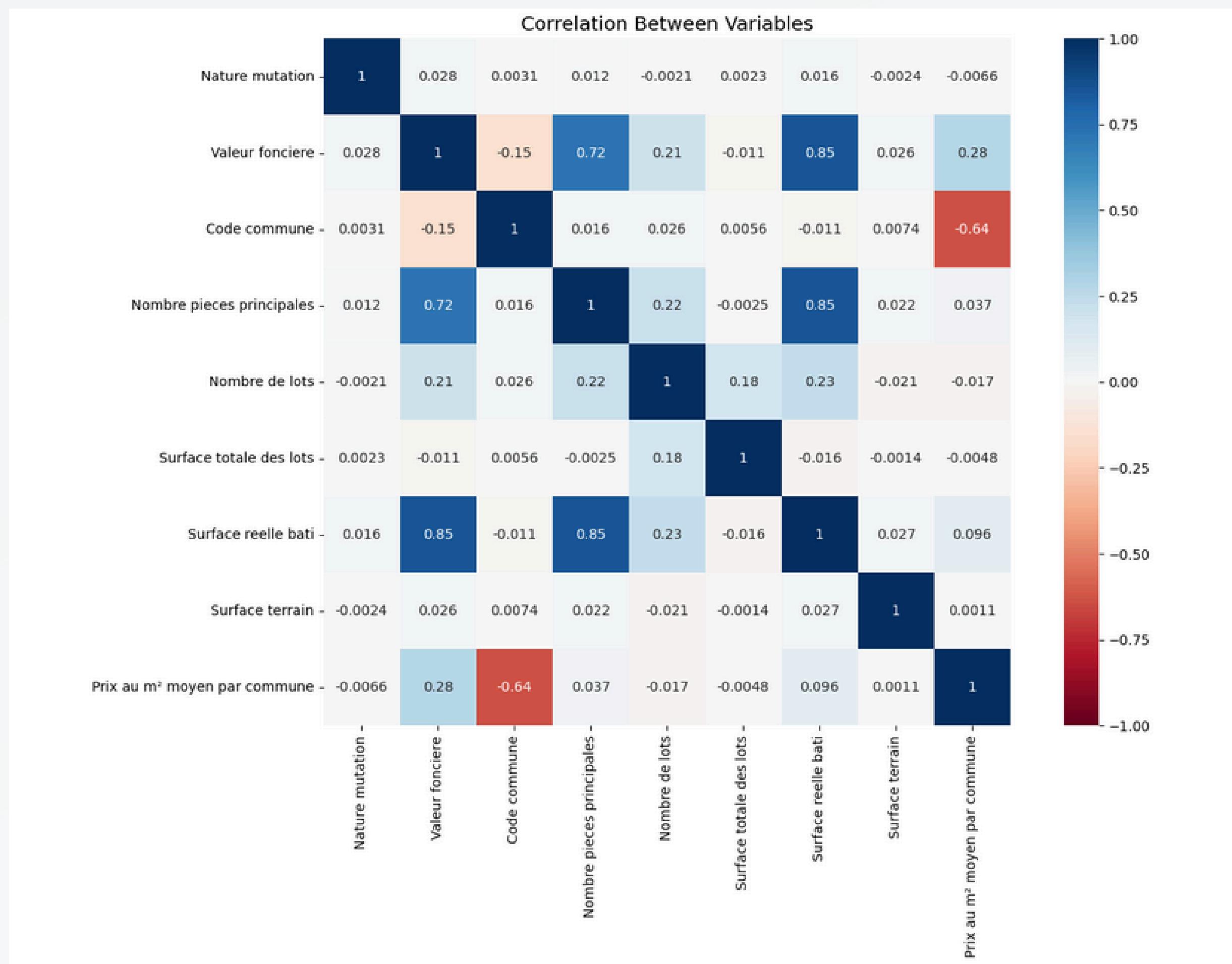


# BOXPLOT

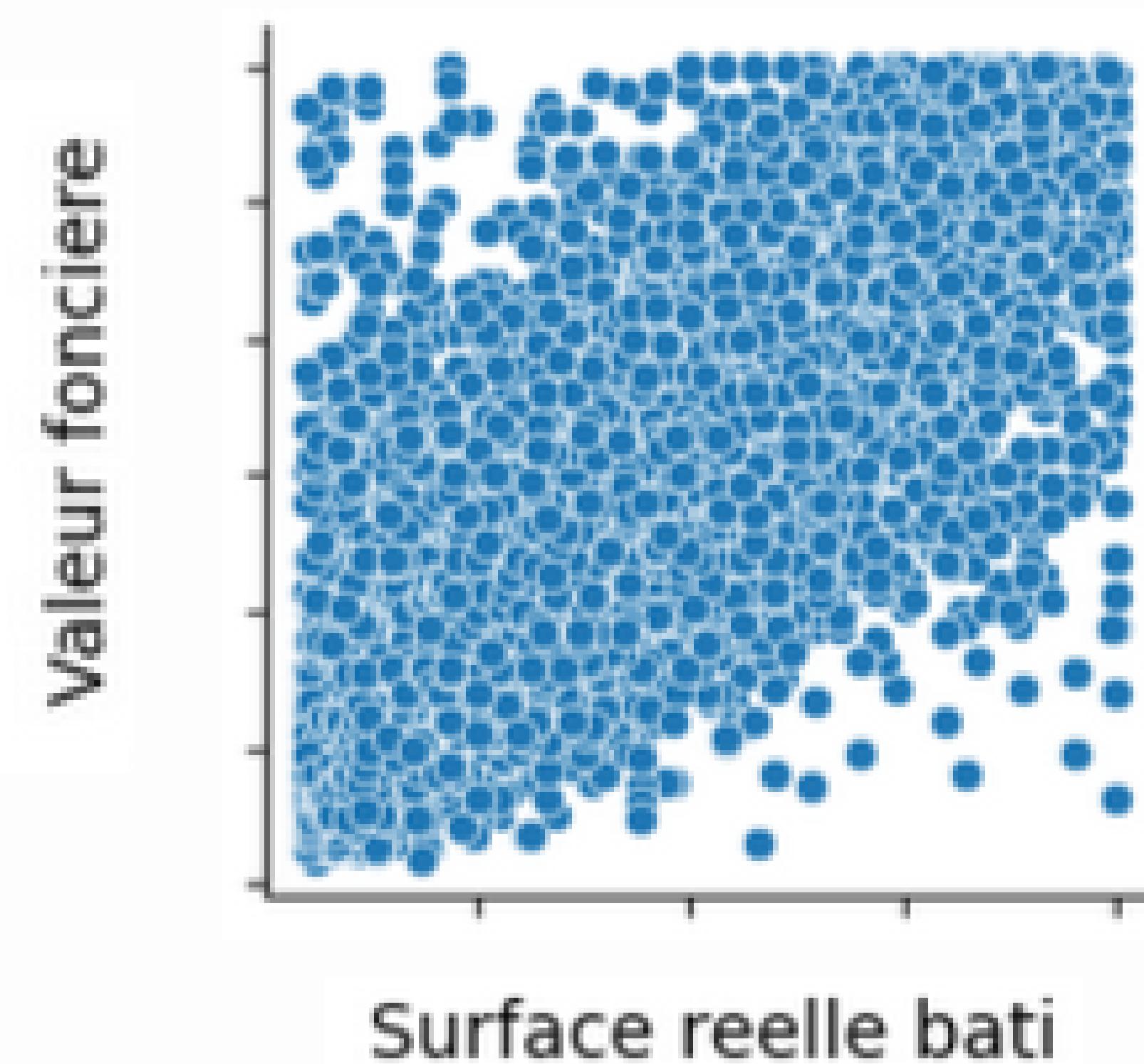
valeur légèrement asymétrique avec quelques valeurs très élevées



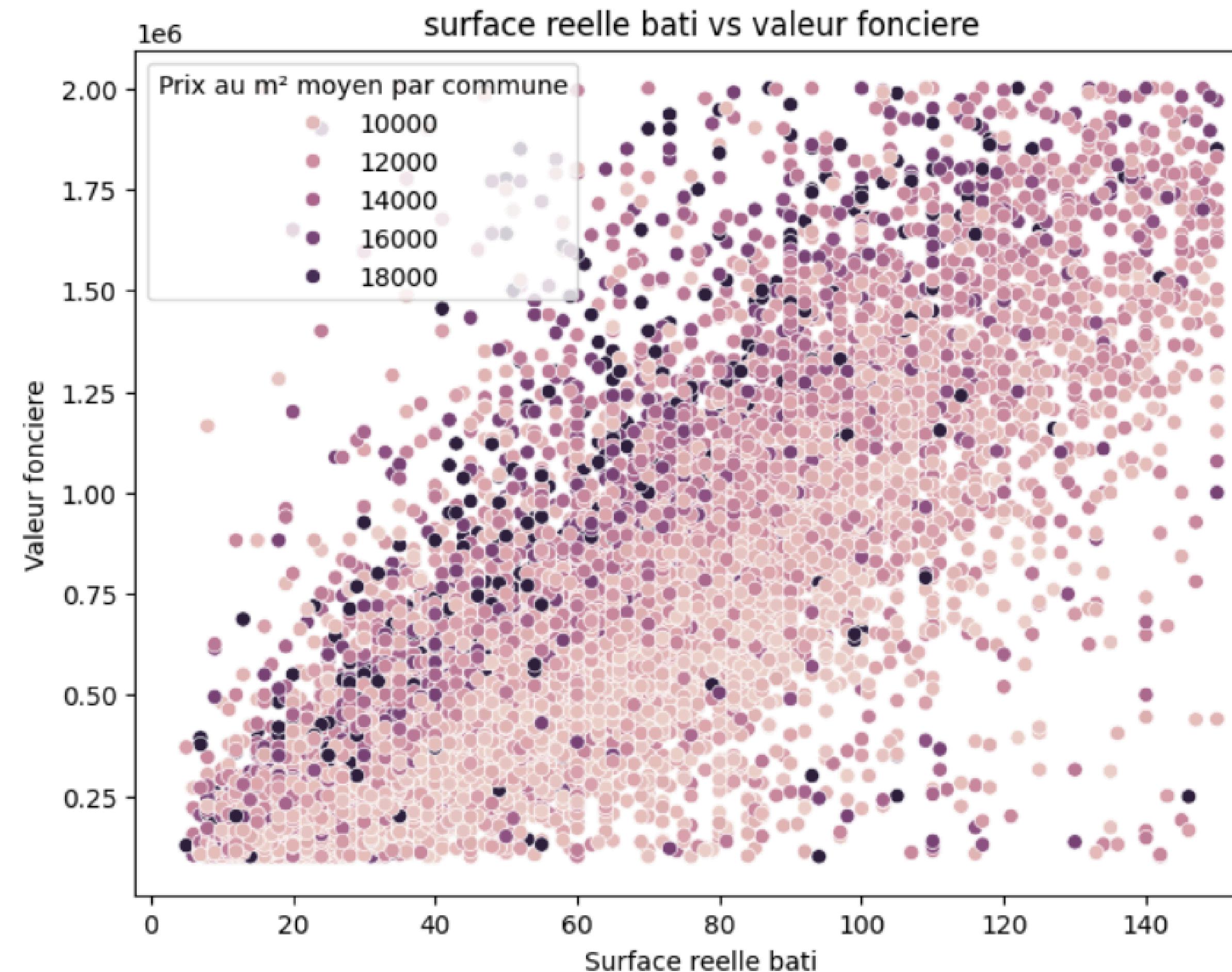
# AFFICHAGE DES DONNEES



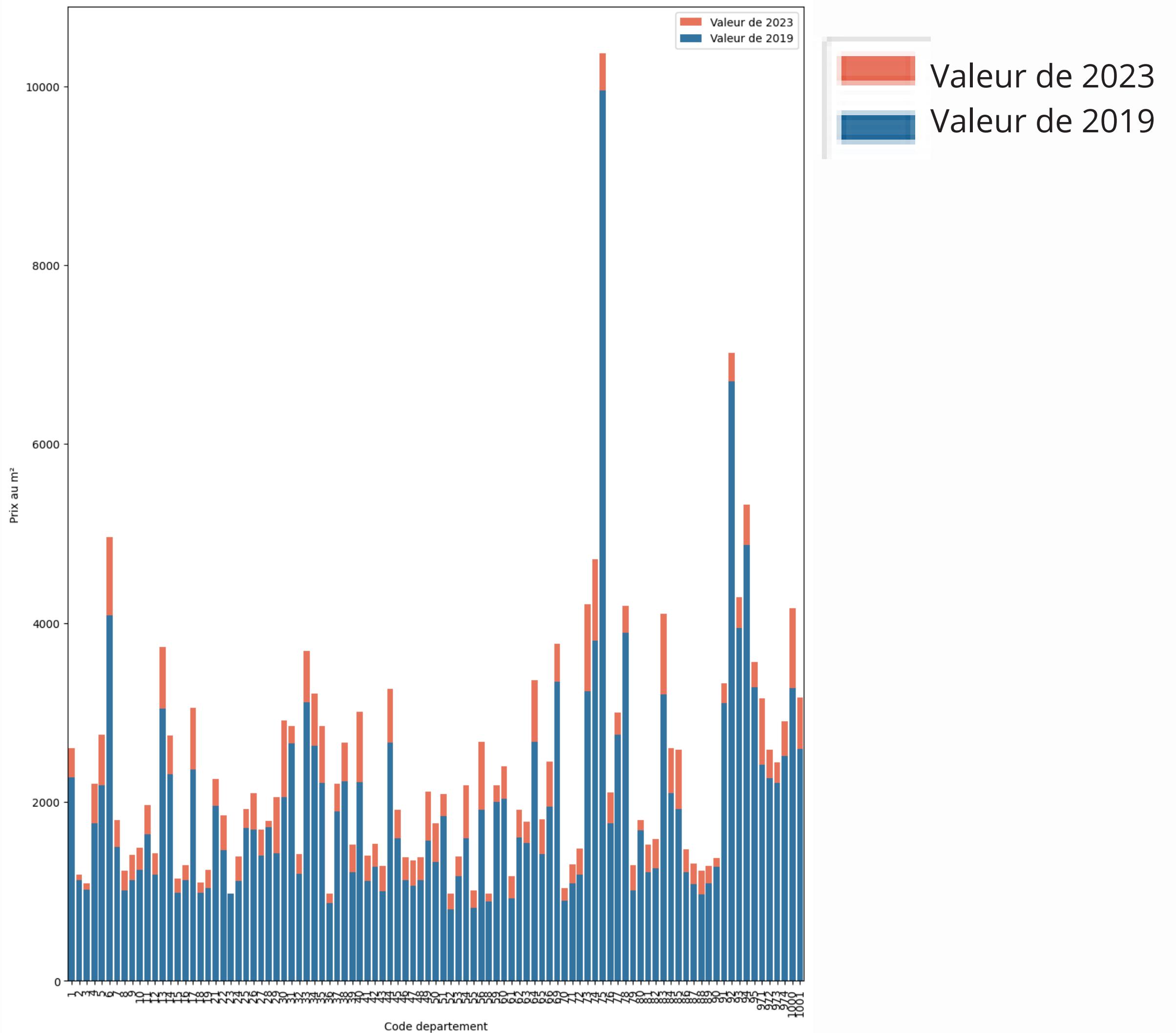
# AFFICHAGE DES DONNEES DANS PARIS



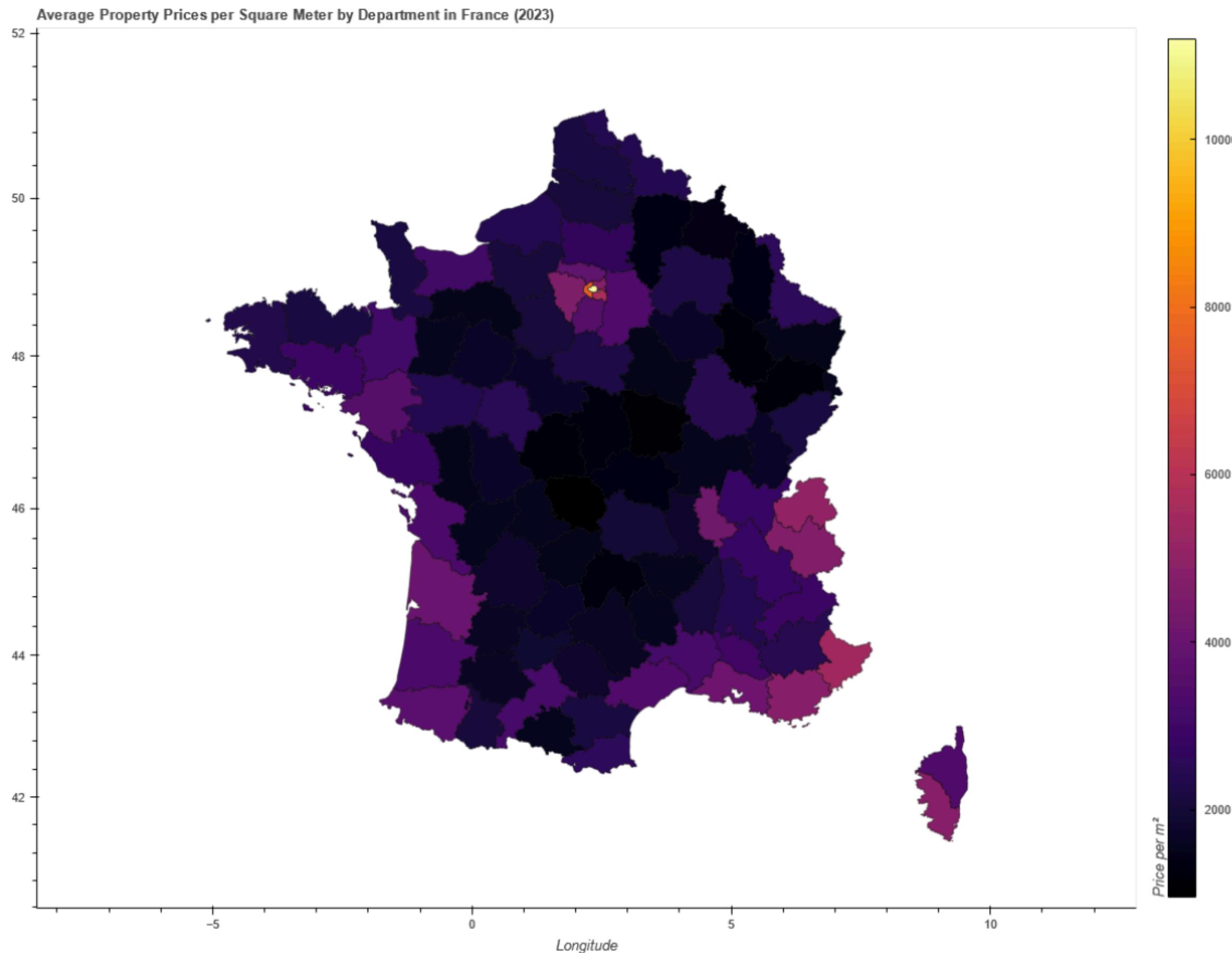
# AFFICHAGE DES DONNEES DANS PARIS



# AFFICHAGE DES DONNEES



# AFFICHAGE DES DONNEES



# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

## NORMALISATION DES DONNÉES

```
normalized_df = StandardScaler().fit_transform(df_data)
```

# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

## CALCUL DE L'ACP

```
n_components = np.argmax(np.cumsum(pca.explained_variance_ratio_) >= 0.90) + 1
```

Nombre optimal de composantes principales : 6

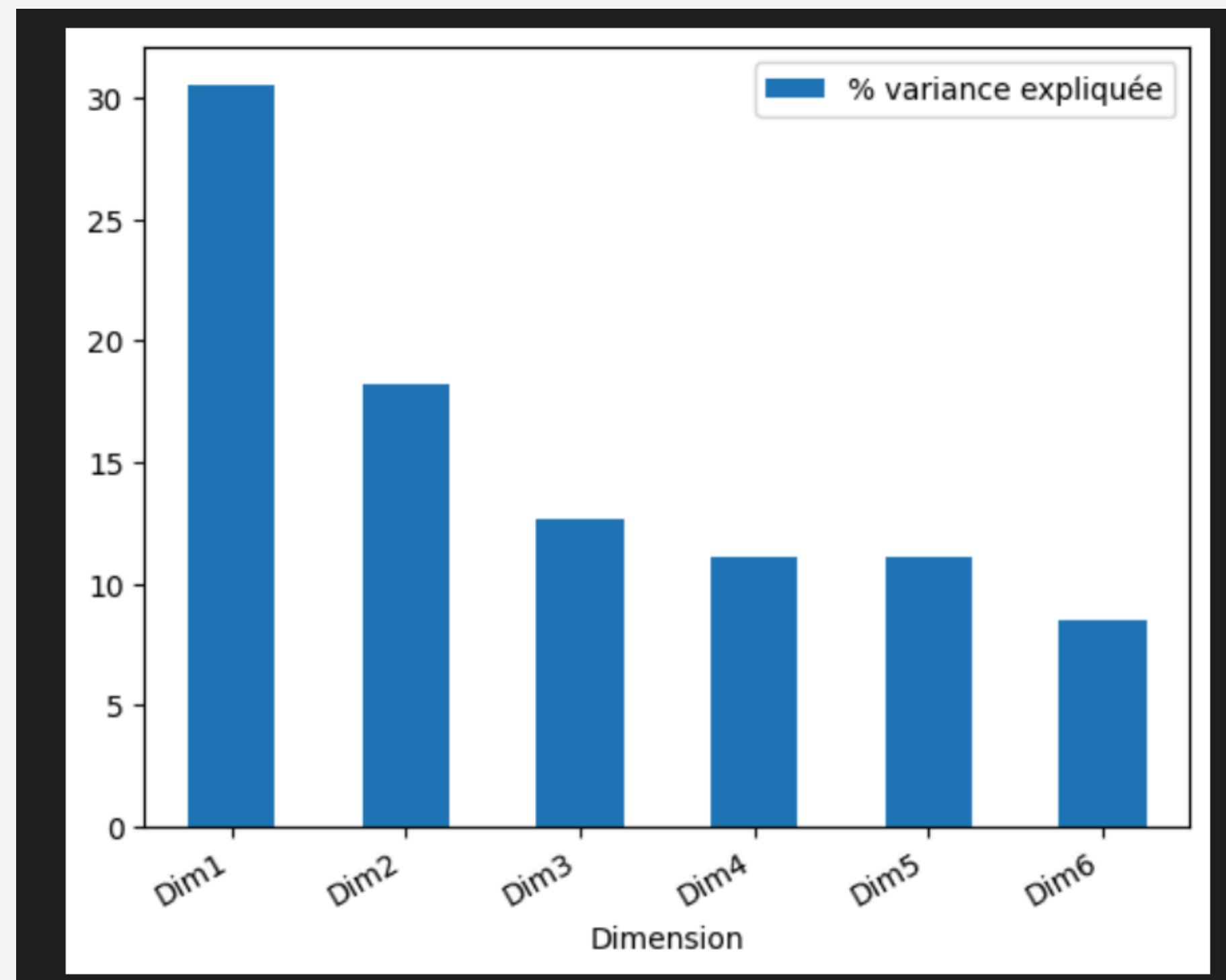
# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

Variance expliquée par cluster

Dimension	Variance expliquée	% variance expliquée	% cum. var. expliquée
0	2.746204	30.51	30.51
1	1.638679	18.21	48.72
2	1.136426	12.63	61.35
3	1.002113	11.13	72.48
4	0.996172	11.07	83.55
5	0.768737	8.54	92.09

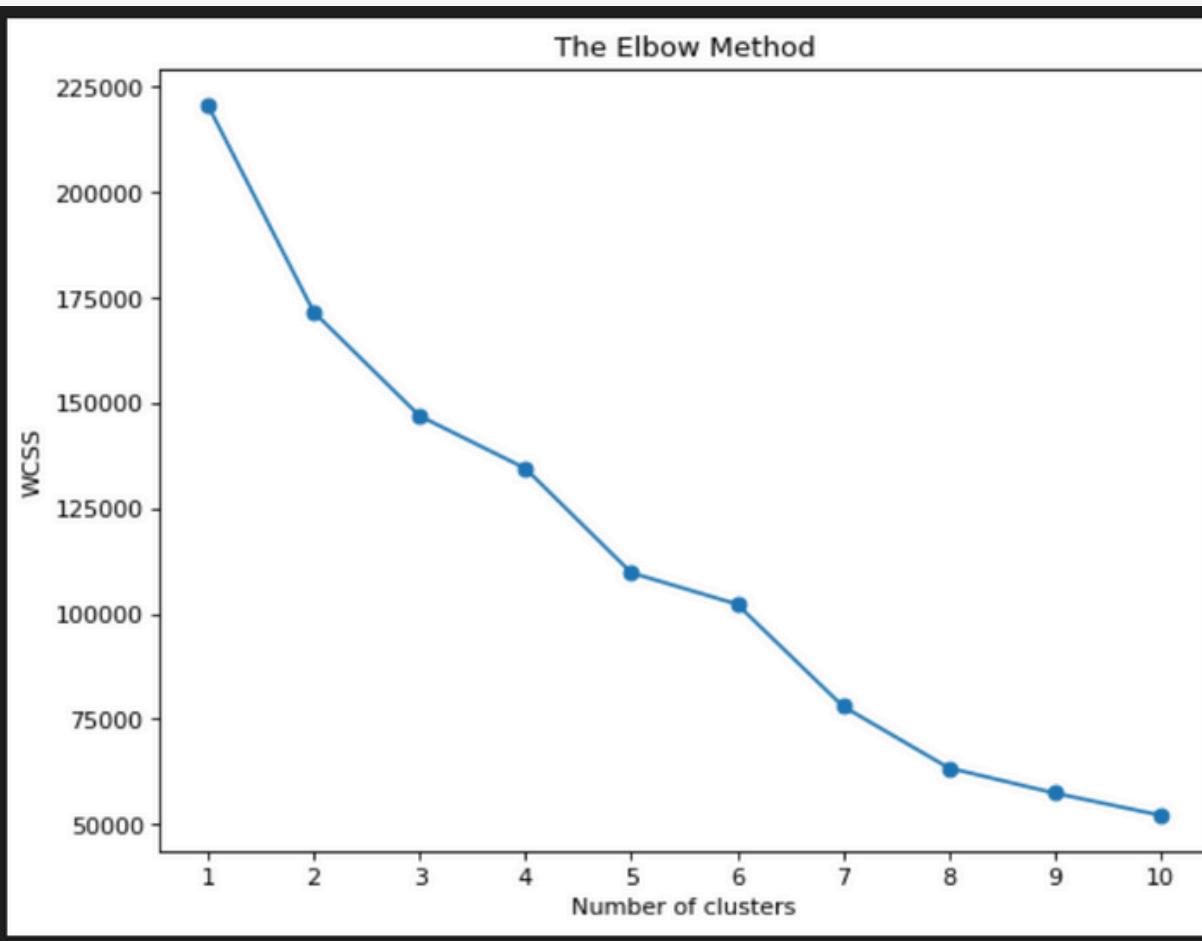
# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

plot des variances expliquées



# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

## ALGORITHME DES K-MEANS



Utilisation de la méthode du coude pour déterminer le nombre de cluster ainsi que KneeLocator

```
KneeLocator(range(1, 11), wcss, curve="convex", direction="decreasing")
```

# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

## ALGORITHME DES K-MEANS

Moyennes des colonnes par cluster

	Nature mutation	Valeur fonciere	Code commune	Nombre pieces principales	Nombre de lots	Surface totale des lots	Surface reelle bati	Surface terrain	Prix au m <sup>2</sup> moyen par commune
0	2.001087	6.626677e+05	115.041608	3.107281	1.734203	0.006996	66.461419	0.071107	10306.682588
1	2.000000	5.078372e+05	104.781493	1.862785	1.509657	0.023551	38.957625	0.009513	14729.087258
2	2.000436	2.759358e+05	114.855504	1.529044	0.999855	0.000000	28.386146	0.016555	10225.883263
3	2.001502	1.279638e+06	111.715359	4.220053	2.055952	0.037086	105.083365	1.684566	12108.650332
4	2.000000	4.391296e+05	113.674419	2.147287	3.620155	67.977674	40.077519	0.000000	10853.600822
5	2.000000	3.352528e+05	115.061268	1.741788	2.159383	0.034125	34.785347	0.000000	10147.289347
6	0.000000	3.127182e+05	113.409091	2.022727	1.613636	0.000000	39.613636	0.750000	11182.030629

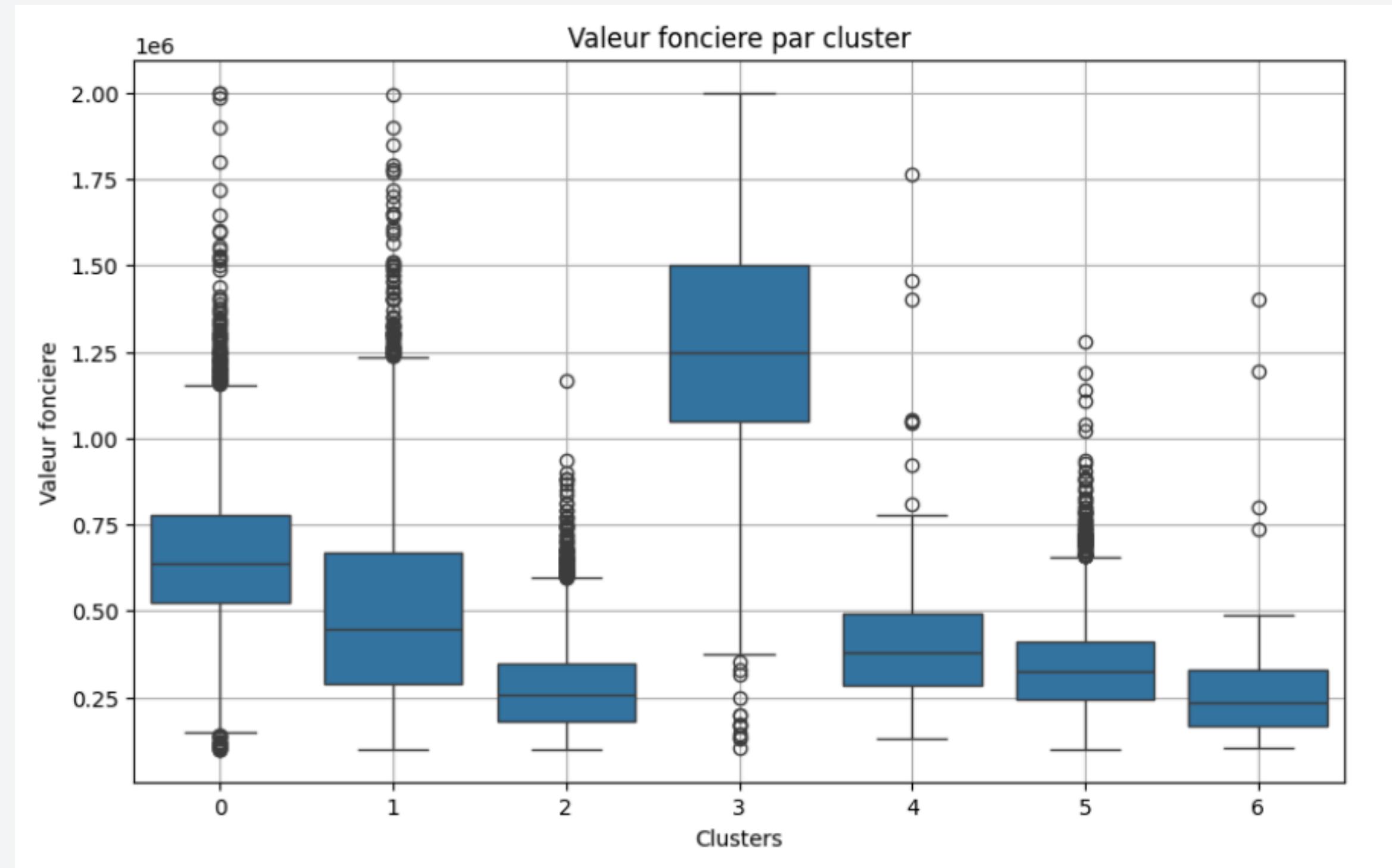
# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

pairplot

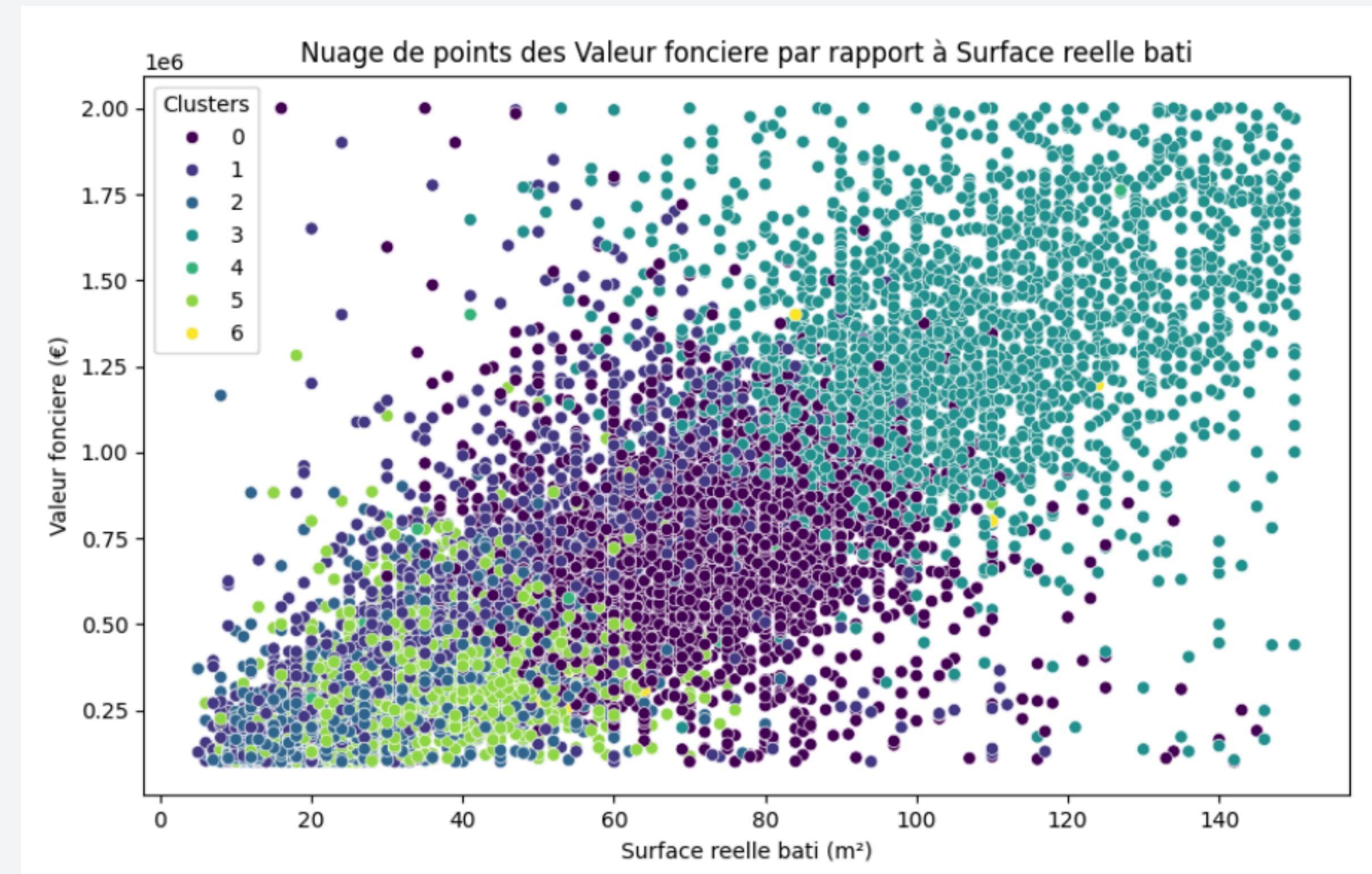


# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

boxplot



# APPRENTISSAGE NON SUPERVISÉ ET CLUSTERING AVEC K-MEANS

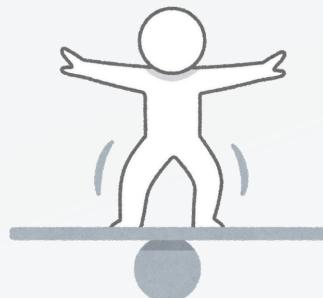


# APPRENTISSAGE SUPERVISÉ: RÉGRESSION LINÉAIRES



*Re-Filtrage des données*

*Suppression des ventes qui avaient un prix au m<sup>2</sup> trop éloigné du prix au m<sup>2</sup>*



*Normalisation des données*

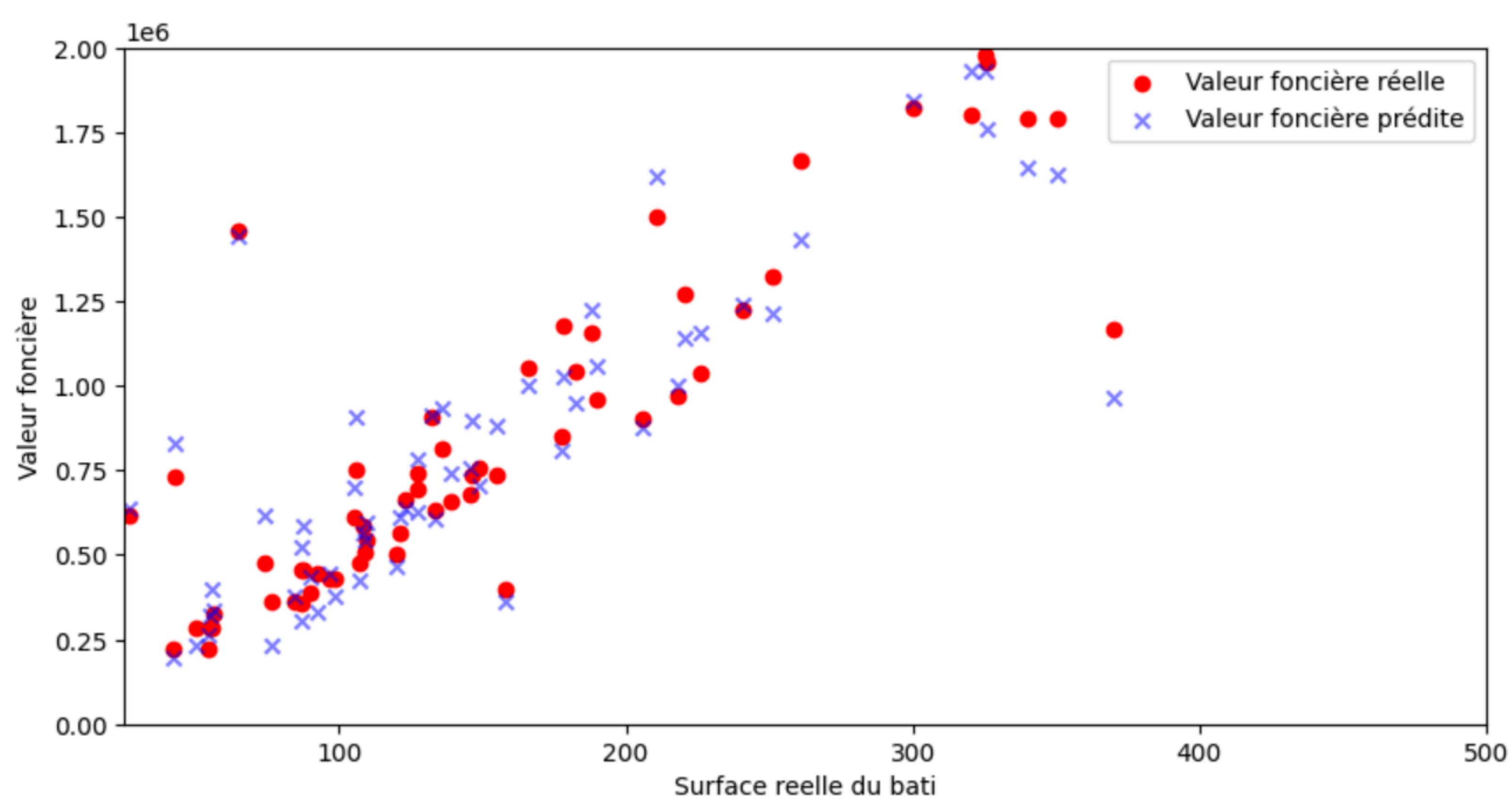
*Utilisation du Standard Scaler de sklearn*



*Régression linéaire par département*

*Choix d'un département précis pour entraîner la régression linéaire de sklearn*

## Département 75:



# APPRENTISSAGE SUPERVISÉ: RÉGRESSION LINÉAIRES

Performance du modèle:

Train MSE: 0.05

Test MSE: 0.04

Train R^2: 0.9480718275768513

Test R^2: 0.962020916493489

Métriques:

$$MSE = \frac{\sum(y_i - \hat{y}_i)^2}{n}$$

$$R2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

# APPRENTISSAGE SUPERVISÉ: RÉGRESSION LINÉAIRES

Paris - 12 ème  
arrondissement

*Et par commune ?*

Train MSE: 0.02

Test MSE: 0.11

Train R^2: 0.9844001685355013

Test R^2: 0.8752579696716604

# CONCLUSION

