Anticipez les besoins en consommation électrique de bâtiments pour la ville de Seattle

Compétences évaluées

- Mettre en place le modèle d'**apprentissage supervisé** adapté au problème métier

- **Transformer** les variables pertinentes d'un modèle d'apprentissage supervisé

- Évaluer les **performances** d’un modèle d'apprentissage supervisé

- Adapter les **hyperparamètres** d'un algorithme d'apprentissage supervisé afin de l'améliorer

Plan

Problématique

I - Cleaning

II – Feature Engineering

III - Exploration

IV – Modélisation

V - Modèle final

Problématique

“Ville neutre en émissions de carbone en 2050
pour les bâtiments non destinés à l’habitation”

- Prédire les émissions de CO2 et la consommation totale d’énergie sans les relevés de consommation annuels coûteux.

- Évaluer l’intérêt de l’ "ENERGY STAR Score" dans la prédiction d’émissions de CO2

Étapes pour la modélisation

11 algorithmes:

- LinearRegression
- Lasso
- Ridge
- Elastic Net
- SVR

Decision Tree
Random Forest

- XGBoost
- AdaBoost
- GradientBoosting

KnearestNeighbors()

2 modèles :
Y= GHGEmissions(MetricTonsCO2e)
Y = SiteEnergyUse(kBtu)

ETAPE 1 – - **Normalisation** des variables numériques
- **Encodage** des variables catégoriques
- **Passage au log** de Y

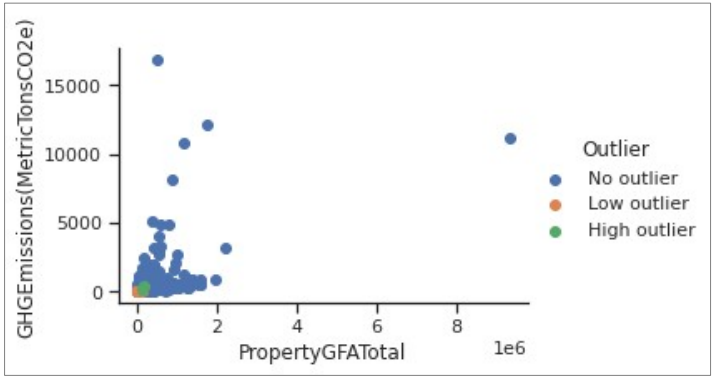
ETAPE 2 – Cross Validation Scoring des modèles

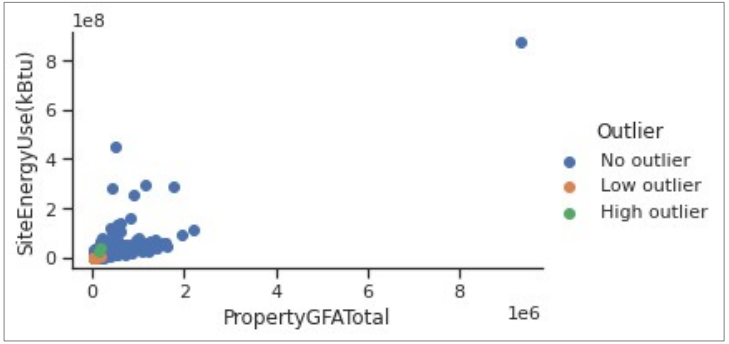
ETAPE 3 - - Évaluation des performances des modèles après entraînement
- Plot de la prédiction ŷ versus valeurs y réelles
- Importance des variables du modèle retenu.

ETAPE 4 - Grid Seach sur les Hyperparamètres du modèle retenu sans ENERGYSTARScore
- Grid Seach sur les Hyperparamètres du modèle retenu avec ENERGYSTARScore

I - Cleaning : traitement des lignes

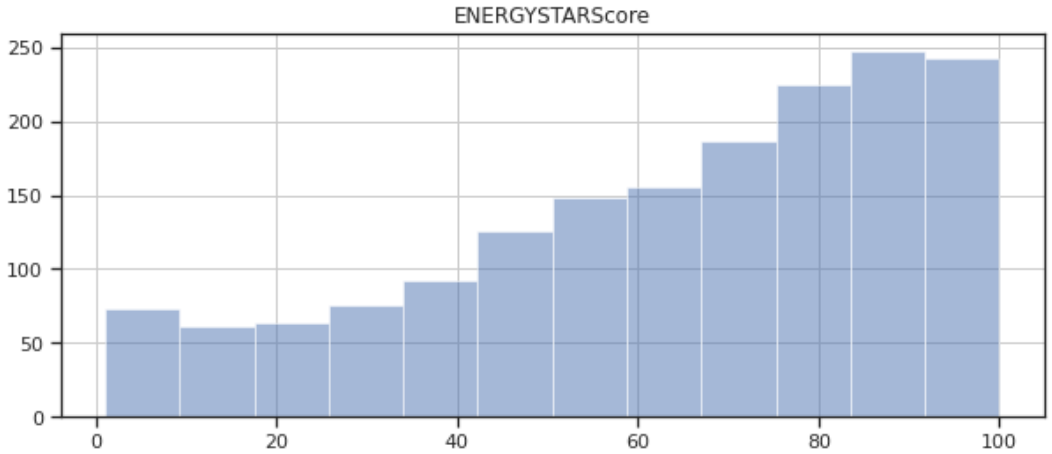
- Suppression des lignes contenant “Multifamily”
- Éviter les **doublons** sur OSEBuildingID
- Suppression de lignes si la valeur est un NaN ou si la valeur tend vers l’infini ou si la valeur est une **valeur aberrante**
- Traitement des NaN avec **KNN Imputer** sur 'ENERGYSTARScore','NumberofPropertyUseTypes' et imputation manuelle sur ZipCode





I - Cleaning : traitement des lignes

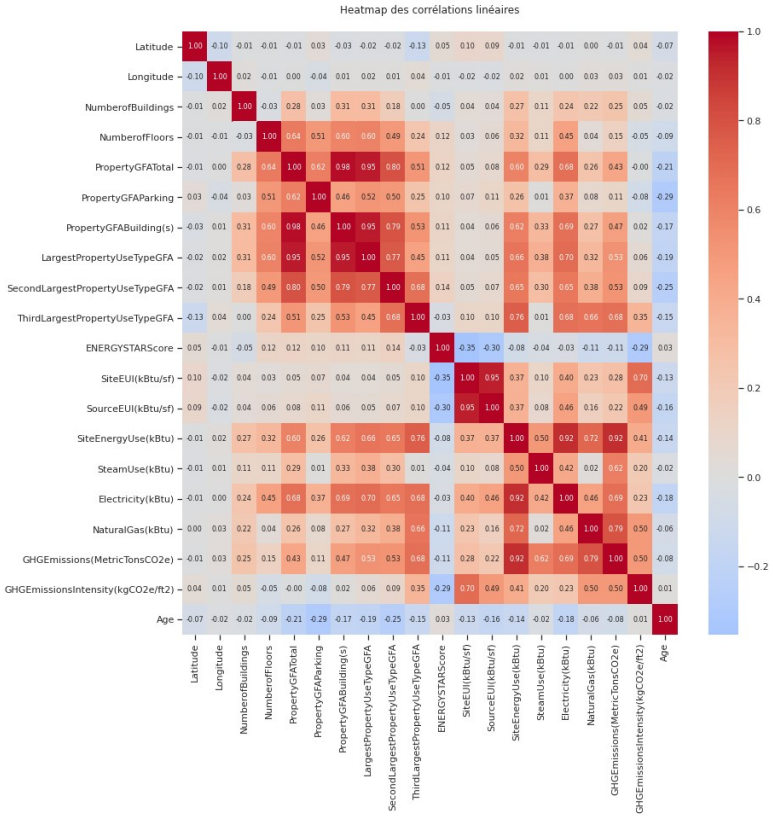
Traitement des NaN avec **KNN Imputer** sur 'ENERGYSTARScore','NumberofPropertyUseTypes' et **imputation manuelle** sur ZipCode



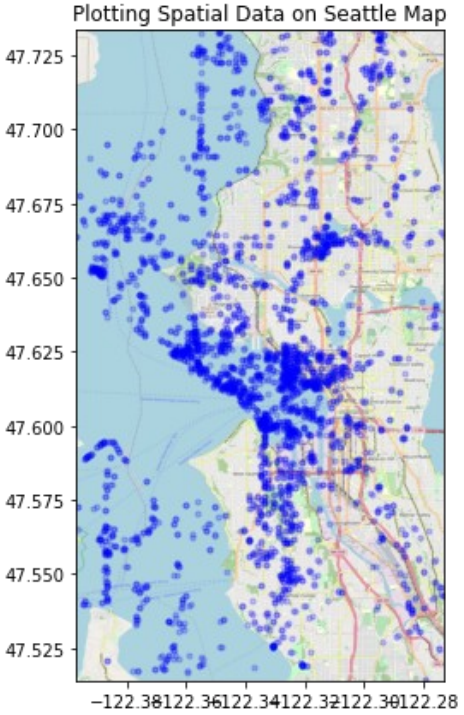
II – Feature Engineering: traitement des colonnes

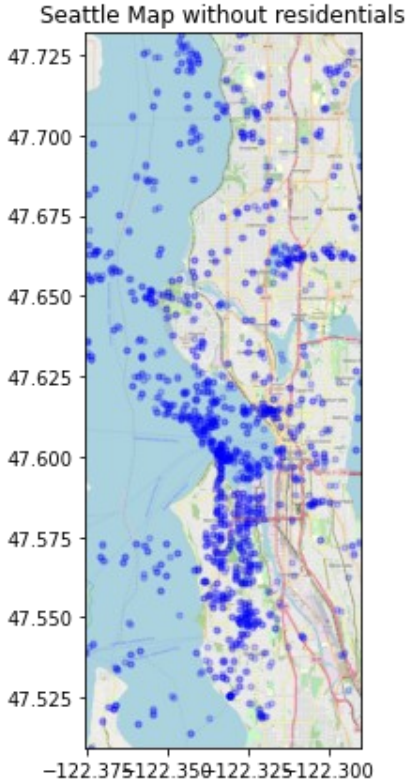
Éviter la **redondance :**

- **Suppression des variables** telles que SteamUse(kBtu), Electricity(kWh), Electricity(kBtu), NaturalGas(therms), NaturalGas(kBtu).
- Suppression des autres colonnes non utiles
- **Création de nouvelles variables** telles que Age, NumberofPropertyUseTypes, GFABuildingRate, GFABuildingRate, GFAPerBuilding, GFAPerFloor, harvesine_distance.



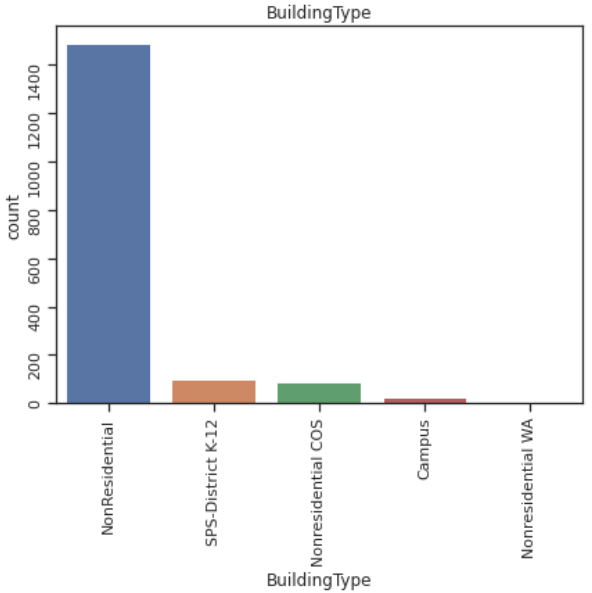
III - Exploration

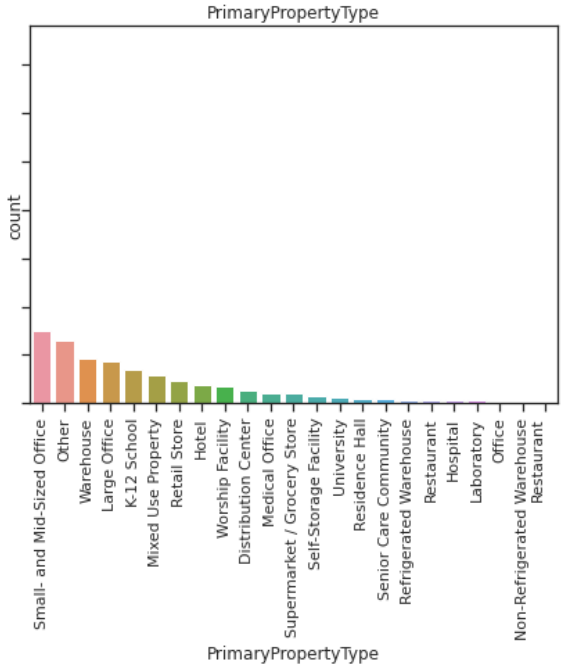




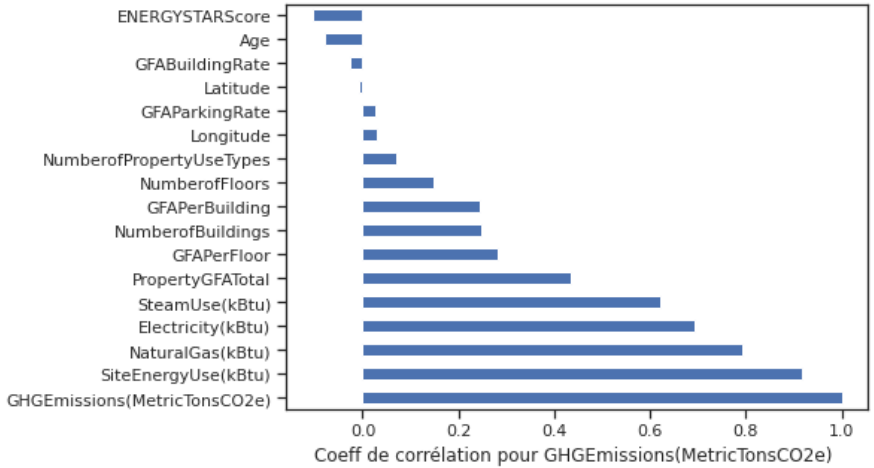
III - Exploration

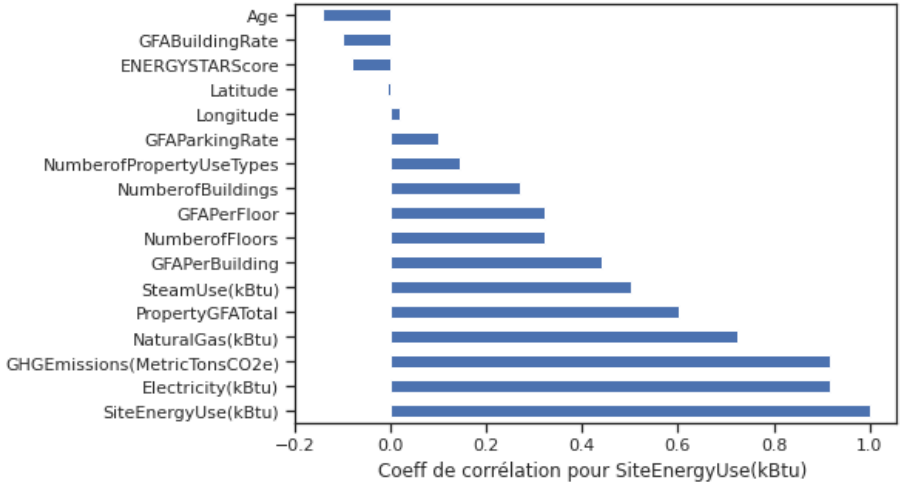
```
Small- and Mid-Sized Office    295
Other                          191
Warehouse                      180
Large Office                   171
Mixed Use Property             104
Retail Store                    94
Hotel                           76
Worship Facility                72
Distribution Center             51
Medical Office                  41
K-12 School                     40
Supermarket / Grocery Store     40
Self-Storage Facility           28
Residence Hall                  21
Senior Care Community           20
University                      17
Refrigerated Warehouse          12
Restaurant                      11
Hospital                        10
Laboratory                      10
Non-Refrigerated Warehouse       2
Restaurant\n                     1
Name: PrimaryPropertyType, dtype: int64
```



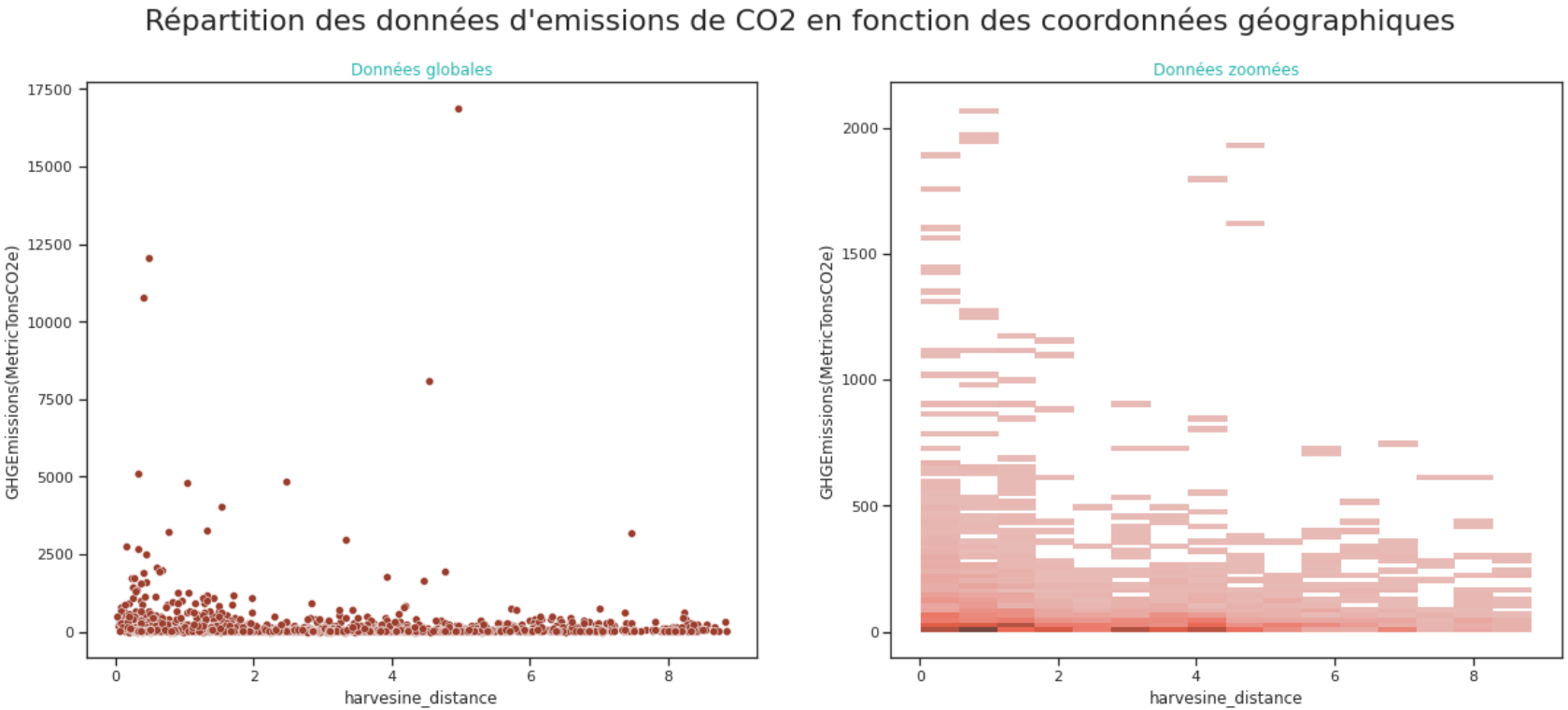


1- Analyse Uni variée – variables numériques



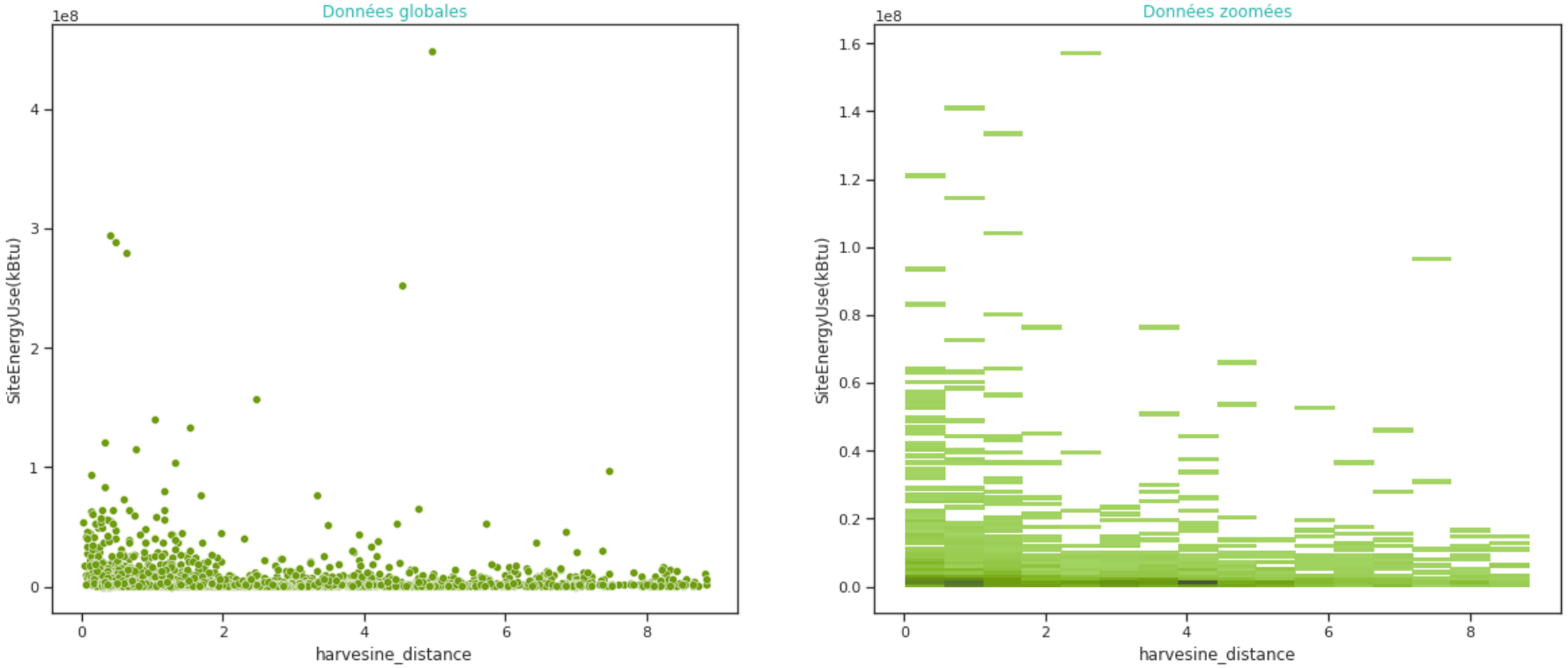


III - Exploration

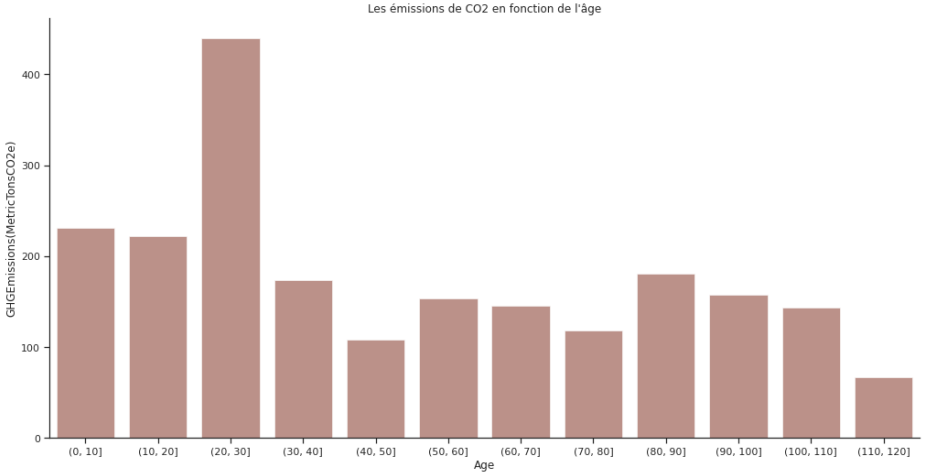


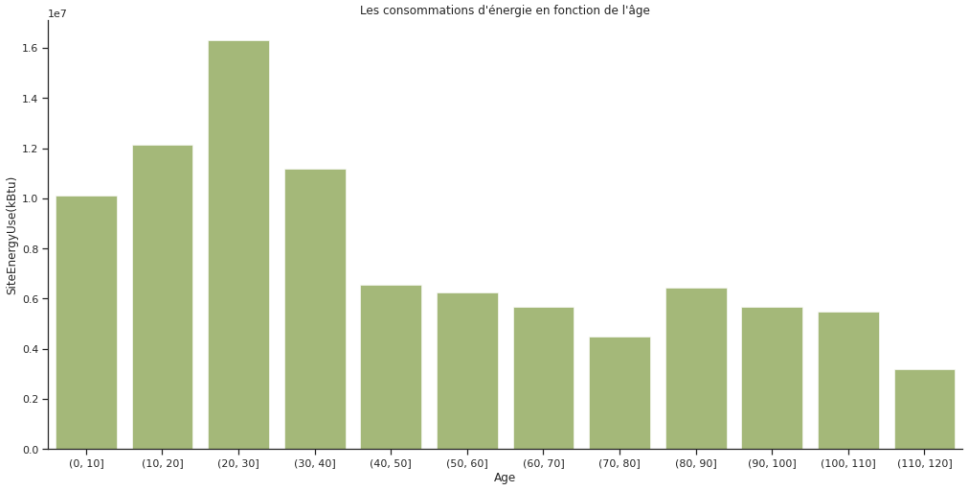
III - Exploration

Répartition des données d'emissions de CO2 en fonction des coordonnées géographiques

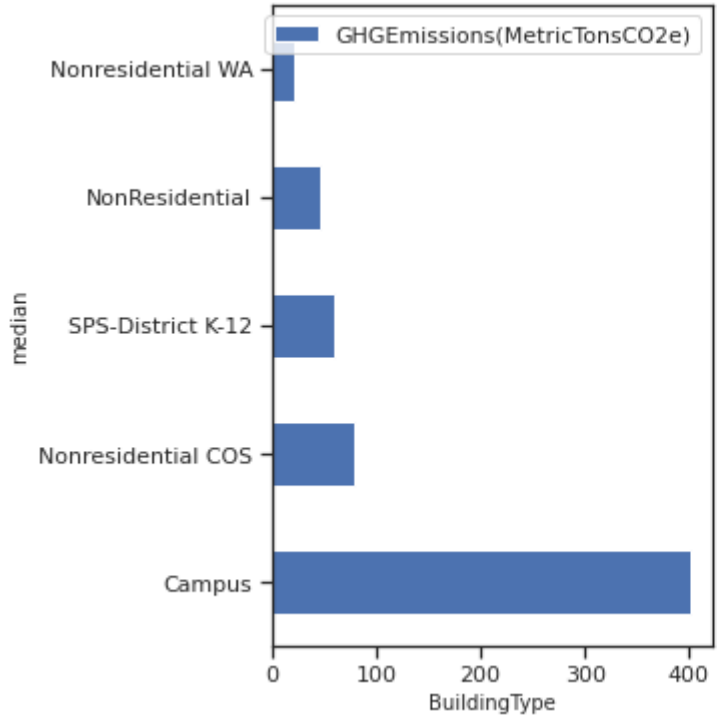


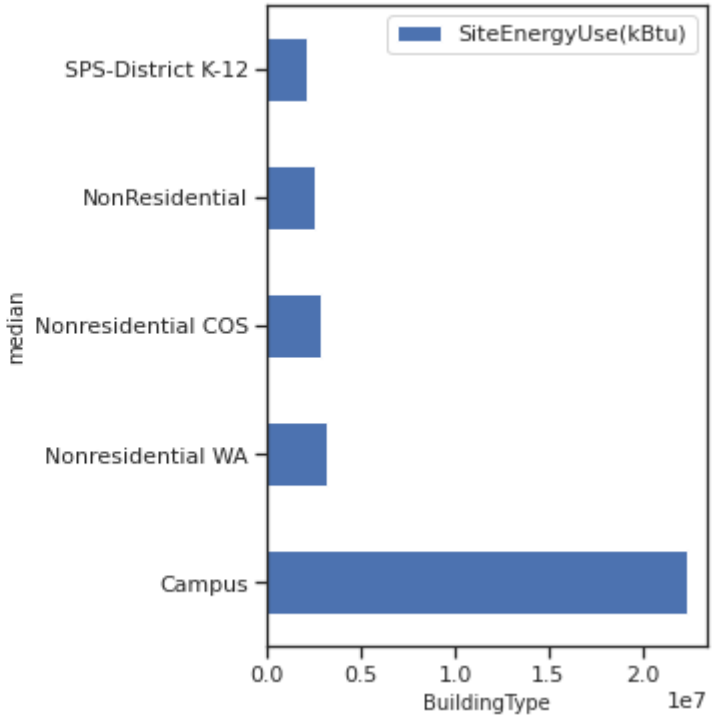
III - Exploration





III - Exploration





IV – Modélisation : Cross Validation Scores

R2 - Coefficient de determination SCE / SCT (ANOVA) compris entre [0,1]

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

y_i la valeur du point i
$\hat{y}_i$ la valeur prédite
$\bar{y}$ la moyenne empirique des points donnés.

MAE - Mean absolute error

$$MAE = \frac{\sum |(\hat{y}_i - y_i)|}{N}$$

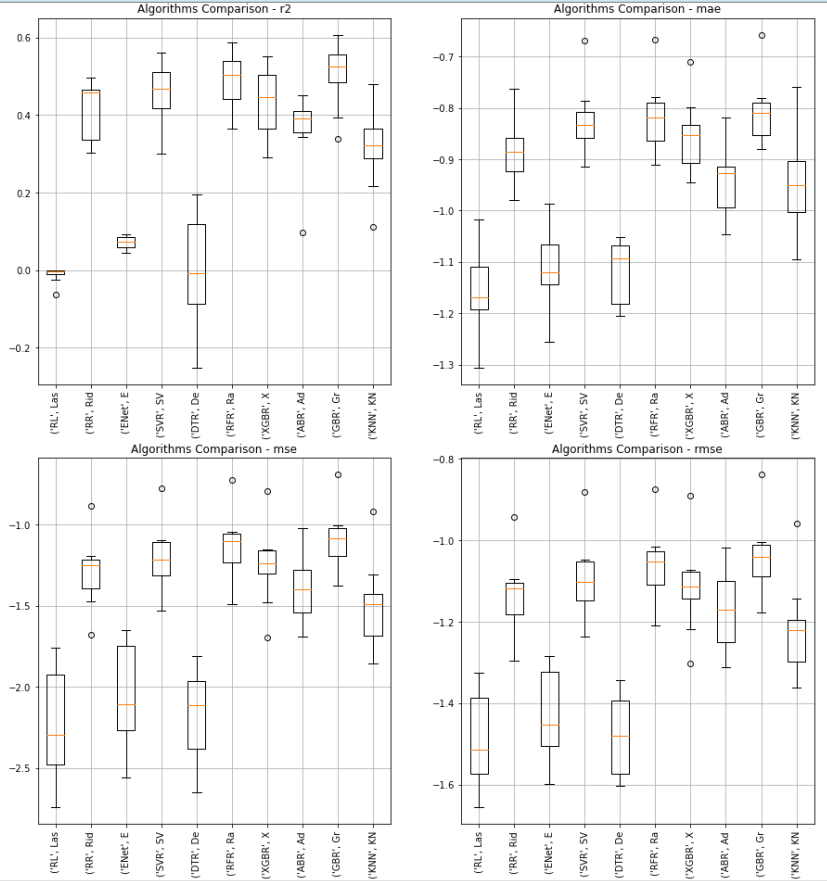
MSE - Mean Squared Error
RMSE - Root Mean Squared Error

$$RMSE = \sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{N}}$$

IV – Modélisation : Cross Validation Scores sur le Train

GHGEmissions(MetricTonsCO2e)

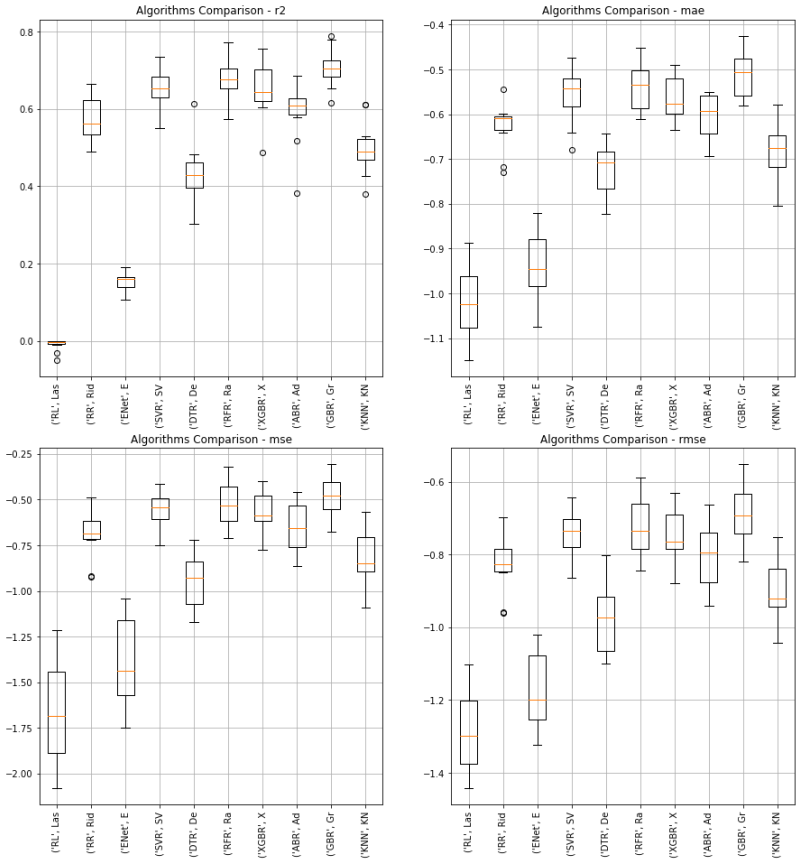
* R2 : **GBR** - Gradient Boosting Regressor
* MAE: **GBR** - Gradient Boosting Regressor
* MSE: **GBR** - Gradient Boosting Regressor
* RMSE: **GBR** - Gradient Boosting Regressor



IV – Modélisation : Cross Validation Scores sur le Train

SiteEnergyUse(kBtu)

* R2 : **GBR** - Gradient Boosting Regressor
* MAE: **GBR** - Gradient Boosting Regressor
* MSE: **GBR** - Gradient Boosting Regressor
* RMSE: **GBR** - Gradient Boosting Regressor



IV – Modélisation : Performance sur le Test

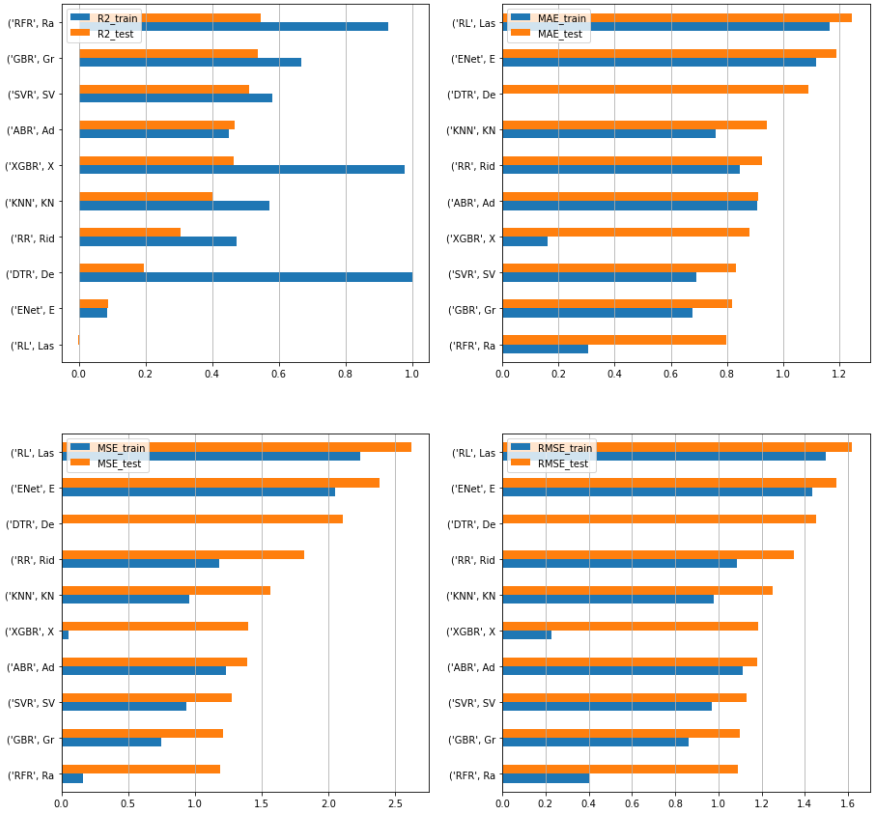
GHGEmissions(MetricTonsCO2e):

* R2 : **ABR** - AdaBoost Regressor car R2_test > R2_train

* MAE : **ABR** - AdaBoost Regressor car l'écart entre le train et test est le plus petit

* MSE : **ABR** - AdaBoost Regressor car l'écart entre le train et test est le plus petit

* RMSE : **ABR** - AdaBoost Regressor car l'écart entre le train et test est le plus petit



IV – Modélisation : Performance sur le Test

SiteEnergyUse(kBtu):

* R2 : **ABR** - AdaBoost Regressor car l'écart entre le train et test est le plus petit tout en ayant un R2 pas trop faible

* MAE : **ABR** - AdaBoost Regressor car l'écart entre le train et test est le plus petit

* MSE : **ABR** - AdaBoost Regressor car l'écart entre le train et test est le plus petit

* RMSE : **ABR** - AdaBoost Regressor car l'écart entre le train et test est le plus petit tout en ayant un RMSE pas trop élevé.

V – Modèle final

En conclusion, même si le GBR obtient de bons résultats, il faut tout de même prendre en condération la **complexité** du modèle rendant celui-ci moins géneralisable aux nouvelles données.

Pour ne pas choisir les extrêmes, j'opterai un modèle qui soit le plus **genéralisable** possible avec des **performances satisfaisantes**.

Nous allons donc séléctionner le modèle AdaBoost Regressor pour prédire la variable SiteEnergyUse et GHGEmissions(MetricTonsCO2e).

V – Modèle final

GridSearch sans EneryStarScore

GHGEmissions(MetricTonsCO2e)

```
explained_variance:  0.4575
r2:  0.4574
MAE:  0.9104
MSE:  1.42
RMSE:  1.1916
```

AdaBoost Regressor sans EneryStarScore

MAE : 0.9133

R2 : 0.4757

RMSE : 1.171

MSE : 1.372

Dans ce cas précis la GridSearch n'est pas nécessaire. On garde alors le premier modèle.

GridSearch + EneryStarScore

```
explained_variance:  0.4592
r2:  0.4592
MAE:  0.9131
MSE:  1.4154
RMSE:  1.1897
```

ENERGYSTARScore améliore que très sensiblement le modèle.

Ce qui corrobore la matrice de corrélation faite dans la première partie : Pélec_01_notebook.

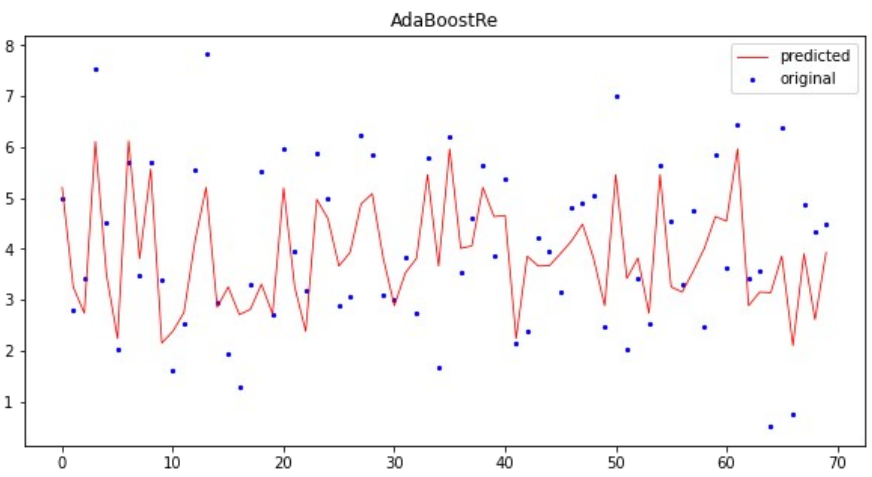
V – Modèle final

GHGEmissions(MetricTonsCO2e)

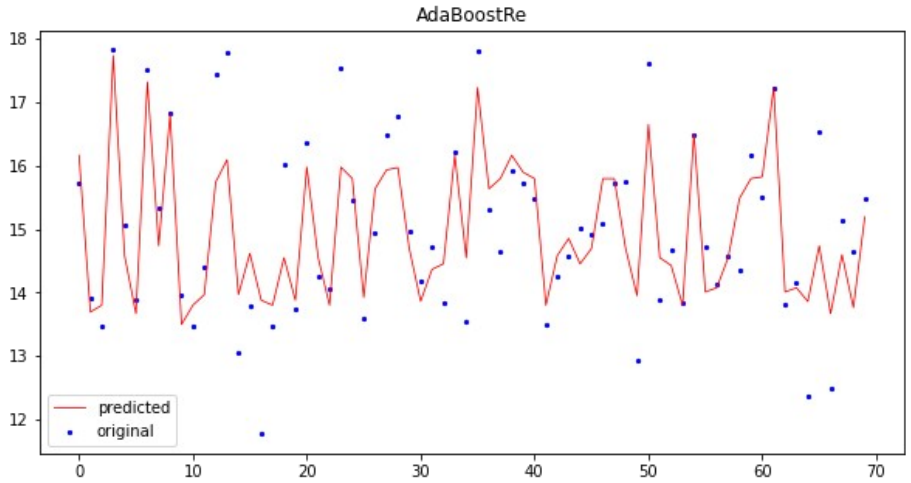
Nous allons donc séléctionner le modèle AdaBoost Regressor sans EneryStarScore et sans GridSearch pour prédire la variable GHGEmissions(MetricTonsCO2e).

Nous allons donc séléctionner le modèle AdaBoost Regressor sans EneryStarScore pour prédire la variable SiteEnergyUse(kBtu).

V – Modèle final

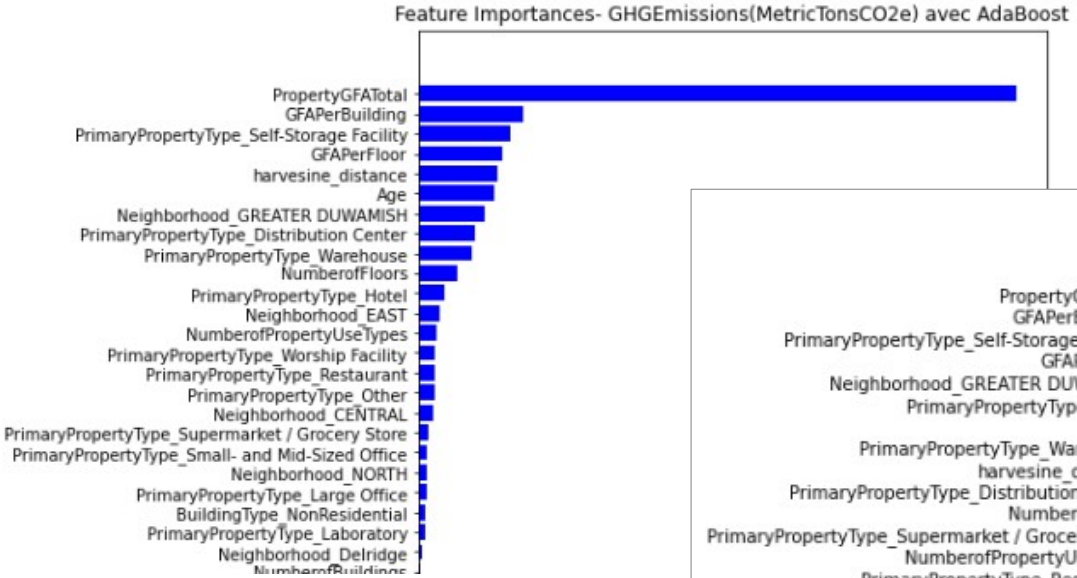


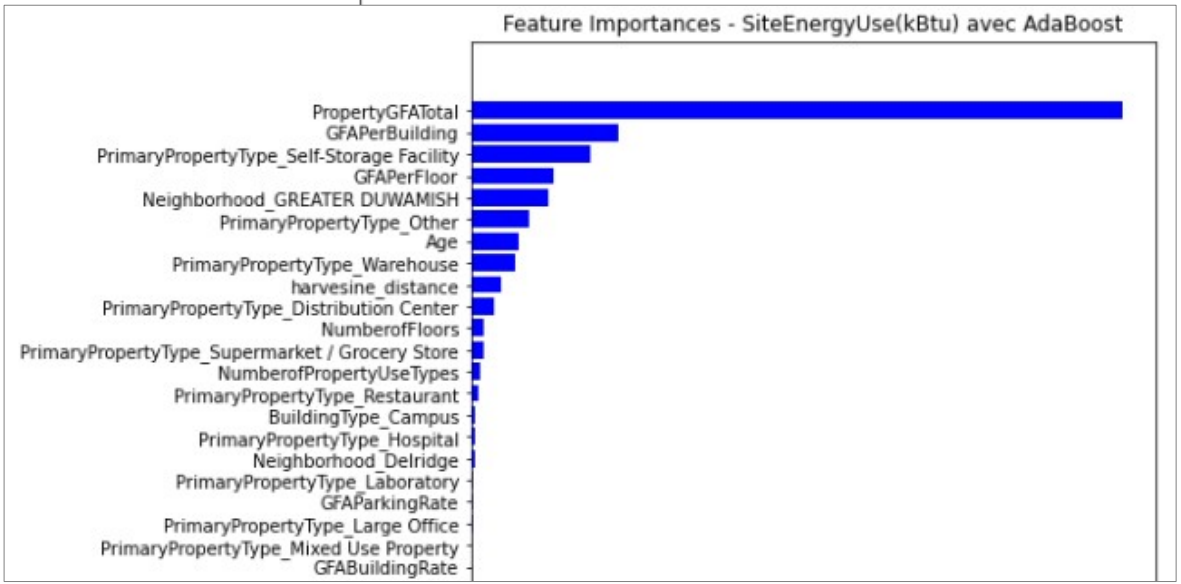
GHGEmissions(MetricTonsCO2e)



SiteEnergyUse(kBtu)

V – Modèle final





MERCI POUR VOTRE ATTENTION