Parcours Ingénieur Machine Learning

Session Mars 2021

OPENCLASSROOMS

Projet 3
Anticipez les besoins en consommation électrique de bâtiments

19/06/2021

Etudiante : QITOUT Kenza

Mentor : Maïeul Lombard

Evaluateur : Souleymane Yattara

CONTEXTE DU PROJET

La ville de Seattle veut atteindre son objectif de ville neutre en émissions de carbone en 2050.

Seattle

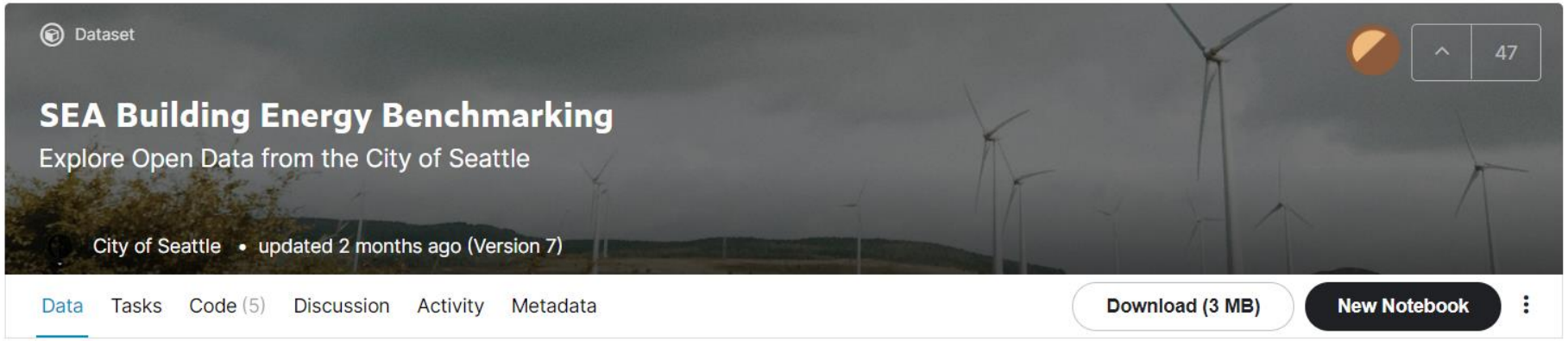
Problématique de la ville de Seattle :

- Relevés coûteux à obtenir
- Donnée "ENERGY STAR Score" fastidieuse à calculer

Objectifs : Prédire les émissions de CO2 et la consommation totale d'énergie et évaluer l'intérêt de l'"ENERGY STAR Score" pour la prédiction d'émissions

BASES DE DONNEES

2 DataFrames de 2015 et 2016 (disponibles sur https://www.kaggle.com/city-of-seattle/sea-building-energy-benchmarking#2015-building-energy-benchmarking.csv)



Base de données disponible à partir d'une plate-forme de données ouverte de la ville de Seattle

Répertorie les informations sur les bâtiments, leur type d'usage, leur surface et les informations sur les émissions de CO2 et la consommation totale d'énergie

Bases de données de 47 variables et 3340 bâtiments en 2015 (fichier .csv de 1.51 Mo) et de 46 variables et 3376 bâtiments en 2016 (fichier .csv de 1.17 Mo)

PISTES DE RECHERCHE

Missions :

- Réaliser une courte analyse exploratoire après avoir nettoyé le jeu de données
- Tester différents modèles de prédiction
- Identifier le modèle final afin de répondre au mieux à la problématique

Méthodologie :

- Chercher à prédire '**SiteEnergyUse**' et '**TotalGHGEmissions**'
- Modélisation en utilisant les caractéristiques des bâtiments
- Avec/Sans ENERGY STAR Score

1. NETTOYAGE ET EXPLORATION

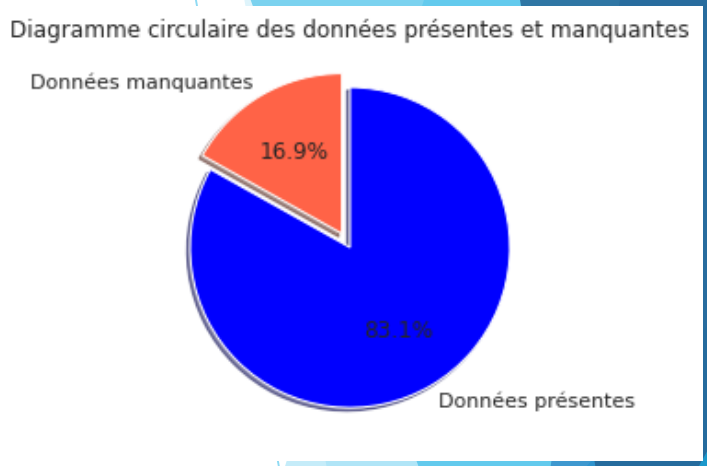
DataFrame de 2015 :

- 10 bâtiments sans données pour 'SiteEnergyUse' et 'TotalGHGEmissions'

⇒ Suppression de 13 bâtiments en construction

⇒ Suppression des 3 lignes avec 'Not Compliant'

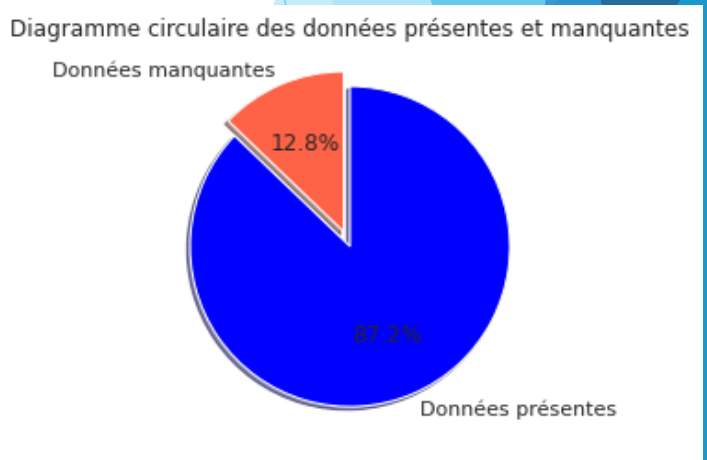
- Variable 'Location‘ pour trouver les informations sur 'Latitude', 'Longitude' et 'ZipCode'



DataFrame de 2016 :

- 9 bâtiments sans données pour 'SiteEnergyUse' et 6 pour 'TotalGHGEmissions'

⇒ Suppression des 37 lignes avec 'Non-Compliant' et des 15 lignes avec 'Missing Data'



Pour les 2 DataFrames :

- Vérification de l’orthographe et transformation des catégories en minuscule pour 'PrimaryPropertyType', 'BuildingType', 'Neighborhood' et des 3 premiers types d’usages

⇒ Suppression lignes sans 'ListOfAllPropertyUseTypes' (133 en 2015 et 6 en 2016)

1. NETTOYAGE ET EXPLORATION

Nettoyage des variables quantitatives identiques :

- Hypothèse : Imputation avec 'LargestPropertyUseType' et 'LargestPropertyUseTypeGFA' (3 en 2015 et 9 en 2016)

→ Suppression des lignes avec des valeurs négatives (5 en 2015 et 1 en 2016)

- Création d'une variable 'tot_energie' = somme des énergies ('SteamUse', 'Electricity', 'NaturalGas', 'OtherFuelUse') uniquement produites hors site

→ Suppression des lignes avec (somme des énergies - 'SiteEnergyUse') < 0 (en 2016, méthode non utilisée car 'OtherFuelUse' non disponible) : 3 lignes

- Vérification de la relation : 'PropertyGFATotal' = 'PropertyGFAParking' + 'PropertyGFABuilding(s)'
- Suppression des lignes lorsque :
  - 'LargestPropertyUseTypeGFA' > 'PropertyGFATotal'
  - 'ThirdLargestPropertyUseTypeGFA' > 'SecondLargestPropertyUseTypeGFA'
  - 'SecondLargestPropertyUseTypeGFA' > 'LargestPropertyUseTypeGFA'

→ 301 lignes en 2015
327 lignes en 2016

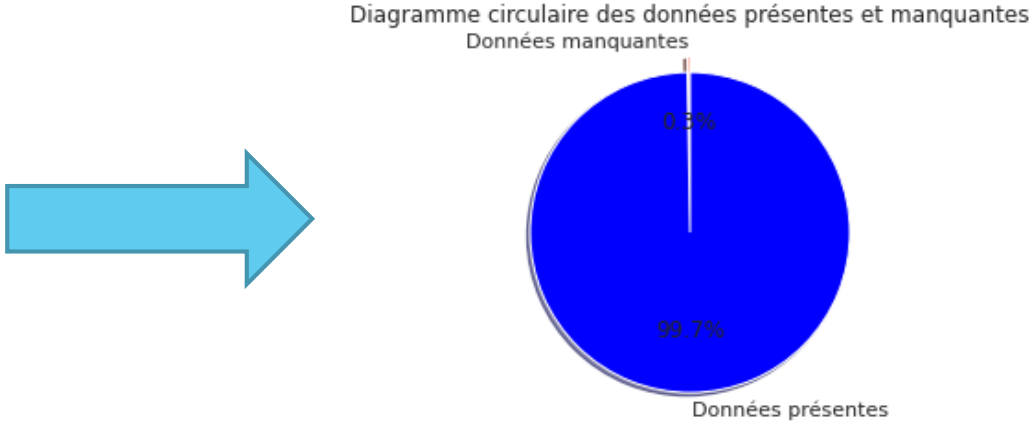
- Choix entre les mêmes variables : unité kBtu et finissant par WN

1. NETTOYAGE ET EXPLORATION

Création de colonnes associées à chaque type d'usage des bâtiments avec *get_dummies* sur les 3 variables LargestPropertyUseType en remplaçant les 1 par la surface du type pondéré par 'PropertyGFATotal' ⇒ 63 nouvelles variables

Création de variables des pourcentages d'énergie : $\frac{énergie}{somme\ totale\ des\ énergies} * 100$

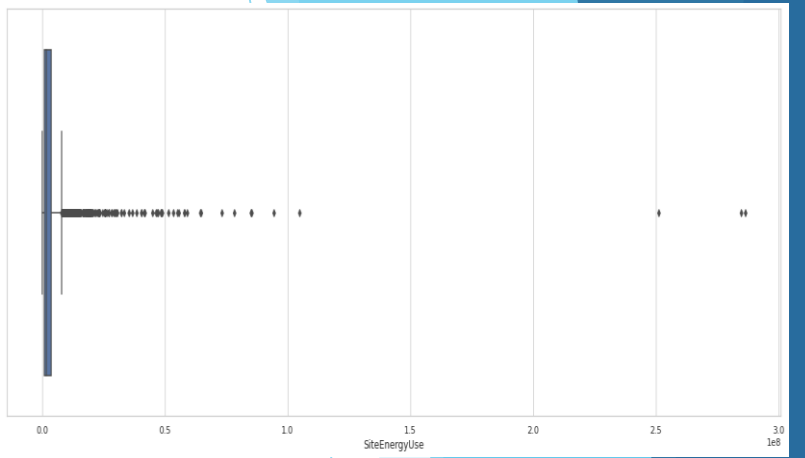
Suppression des colonnes inutiles, Renommer les variables et Vérification des **Outliers** (avec les boxplot) => pas de traitement nécessaire



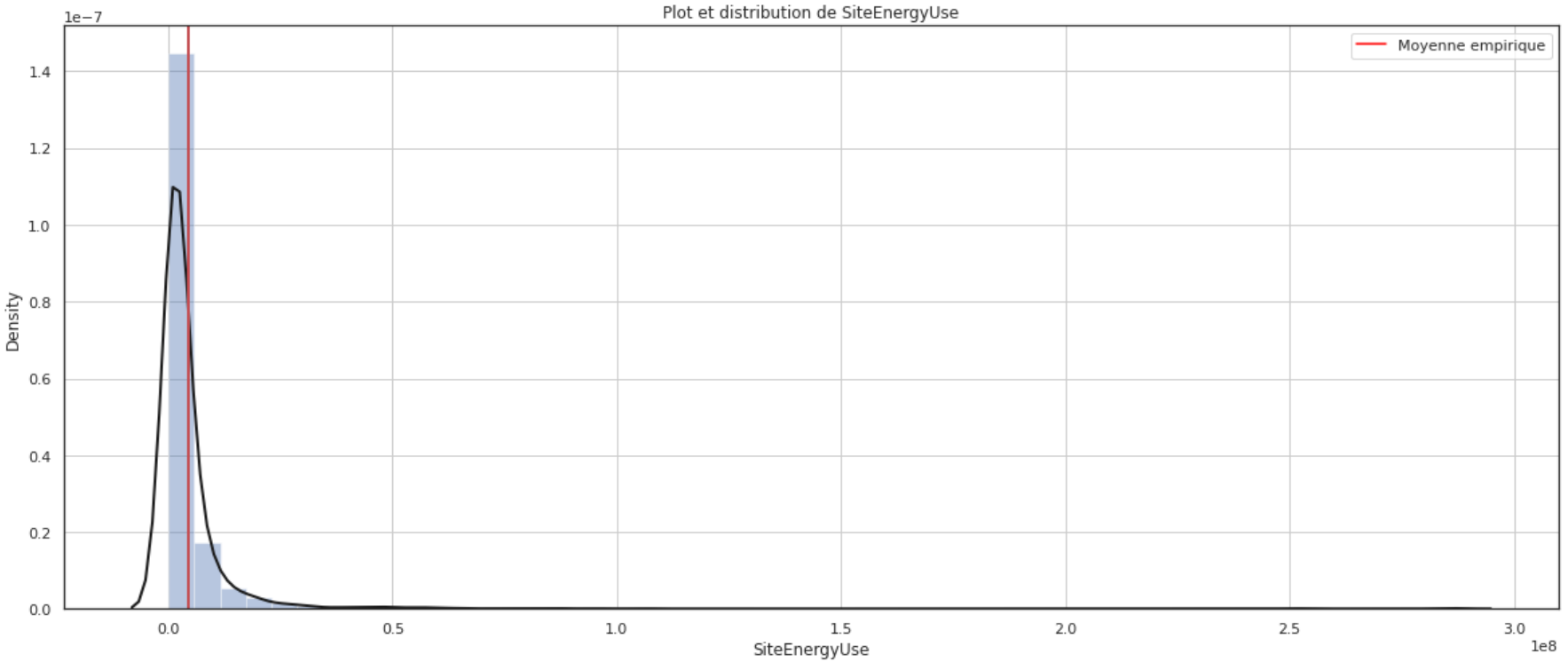
DataFrame final de 2015 = 85% du jeu de données initial avec 477 lignes supprimées

DataFrame final de 2016 = 88% du jeu de données initial avec 389 lignes supprimées

1. NETTOYAGE ET EXPLORATION



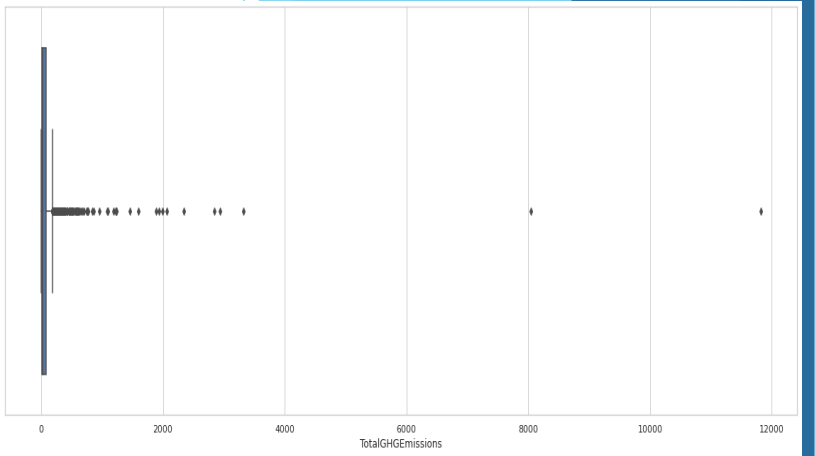
Analyse univariée de 'SiteEnergyUse'

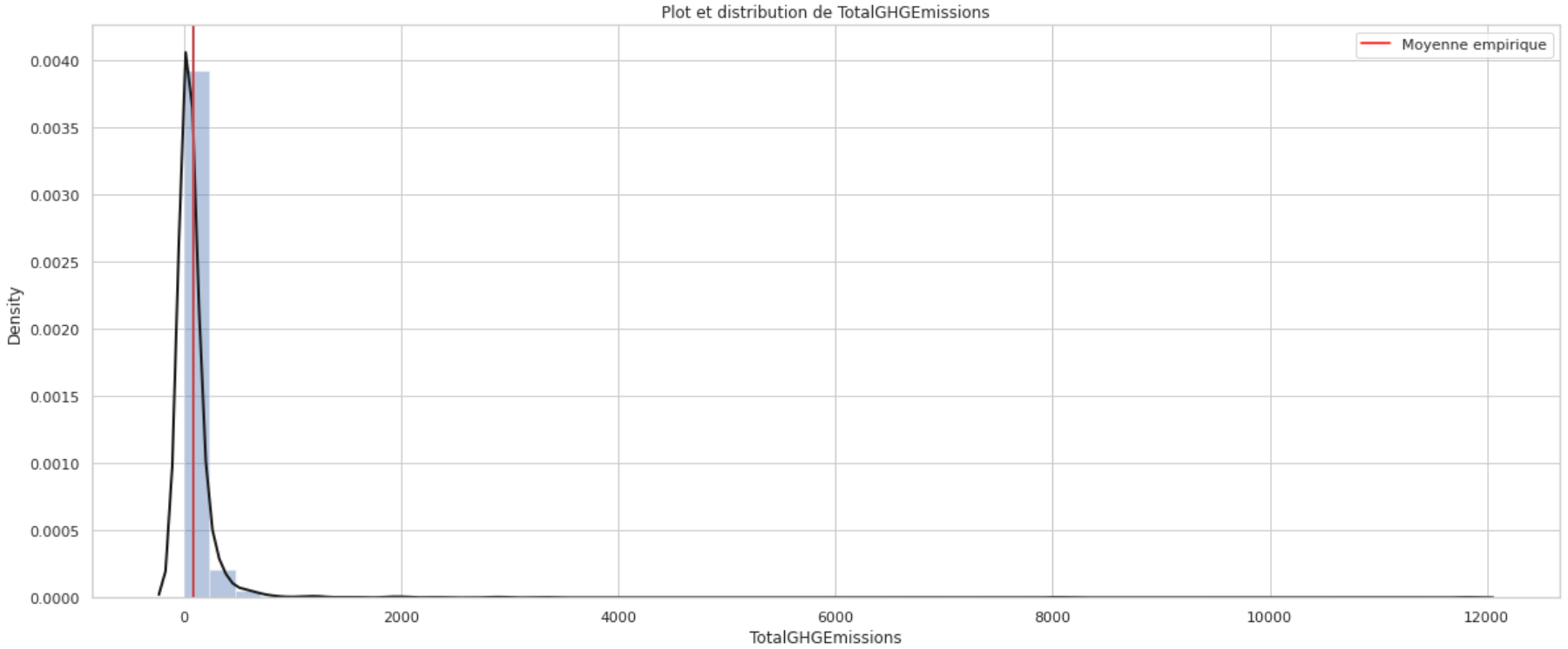


Distribution unimodale
Moyenne : 4673592.89
Médiane : 1692587.0
Ecart-type : 12909623.34
Skewness : 12.31
Kurtosis : 220.78

1. NETTOYAGE ET EXPLORATION

Analyse univariée de 'TotalGHGEmissions'

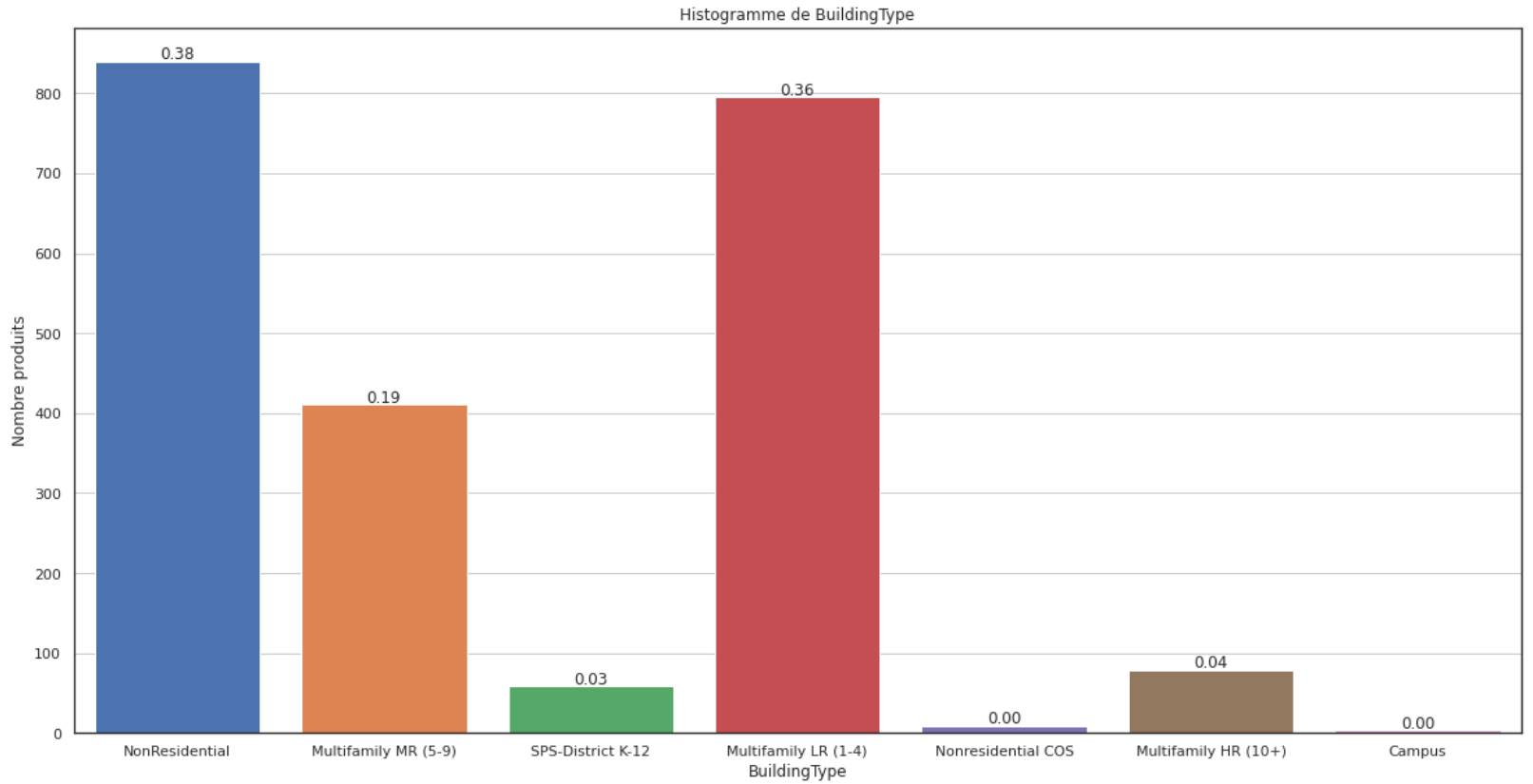




Distribution unimodale
Moyenne : 98.99
Médiane : 30.46
Ecart-type : 359.88
Skewness : 18.72
Kurtosis : 500.65

1. NETTOYAGE ET EXPLORATION

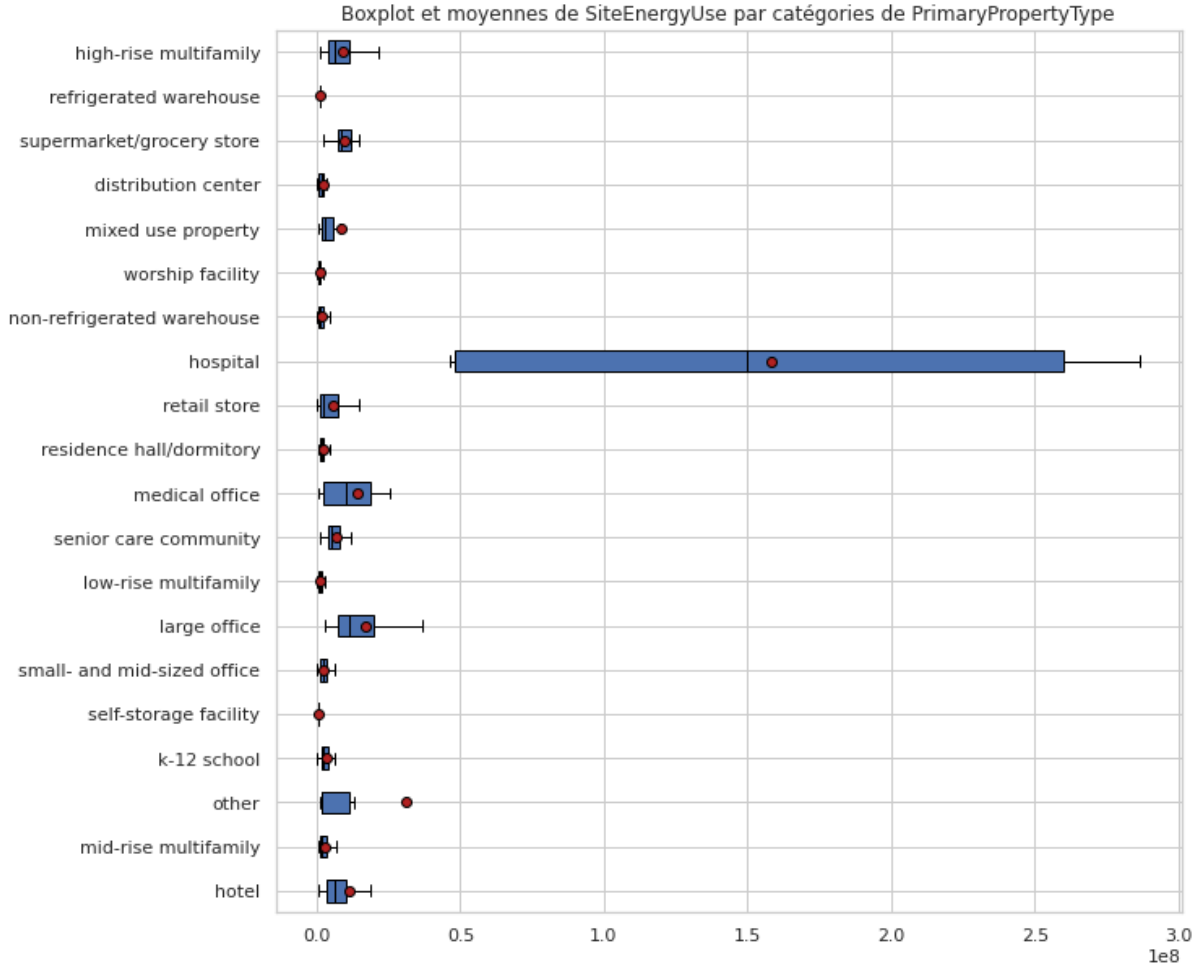
Analyse univariée de 'BuildingType'



7 catégories des types de bâtiments avec la catégorie ‘NonResidential’ la plus fréquente

1. NETTOYAGE ET EXPLORATION

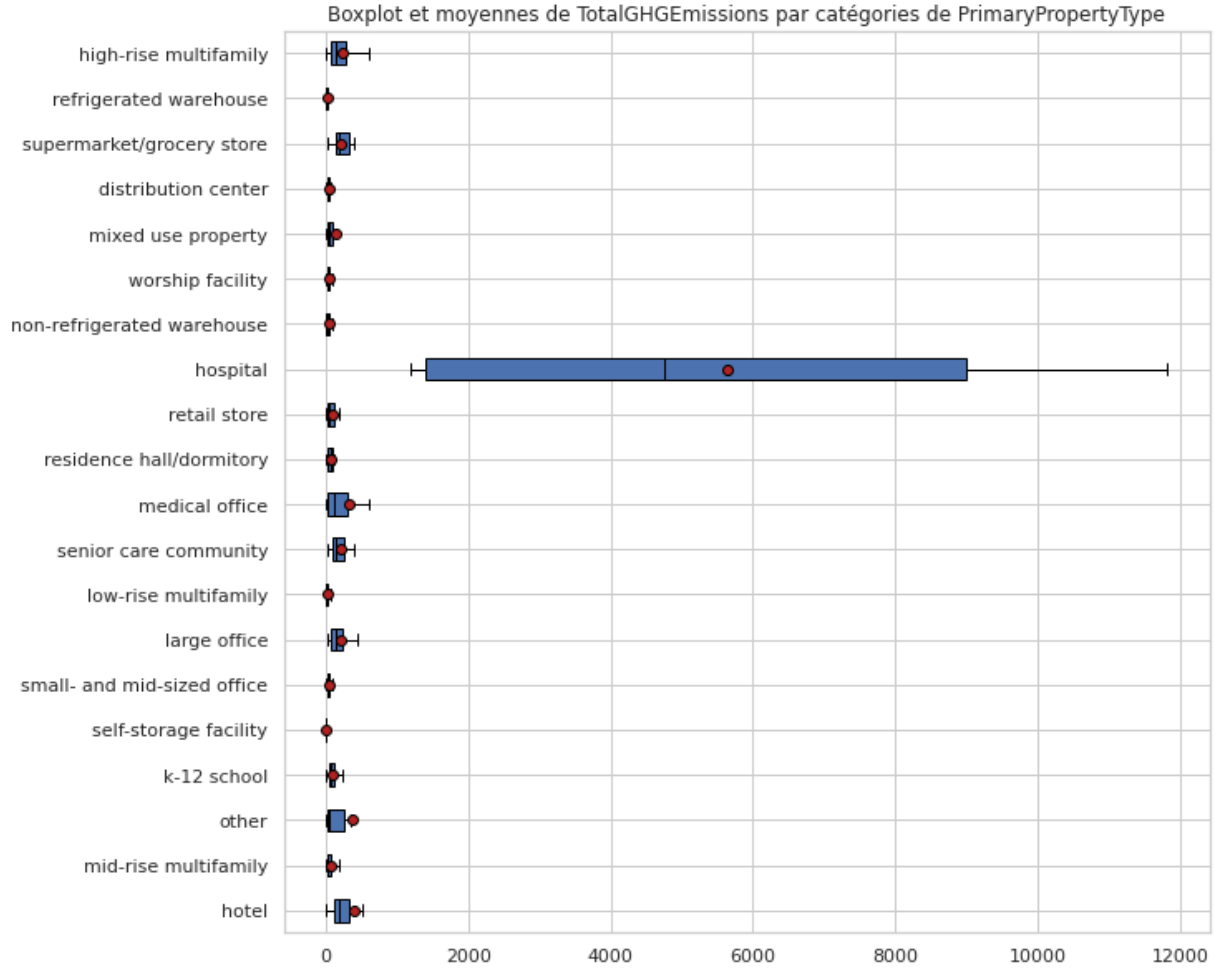
Relation entre 'SiteEnergyUse' et 'PrimaryPropertyType'



ANOVA : $n^2 = 0.41$

1. NETTOYAGE ET EXPLORATION

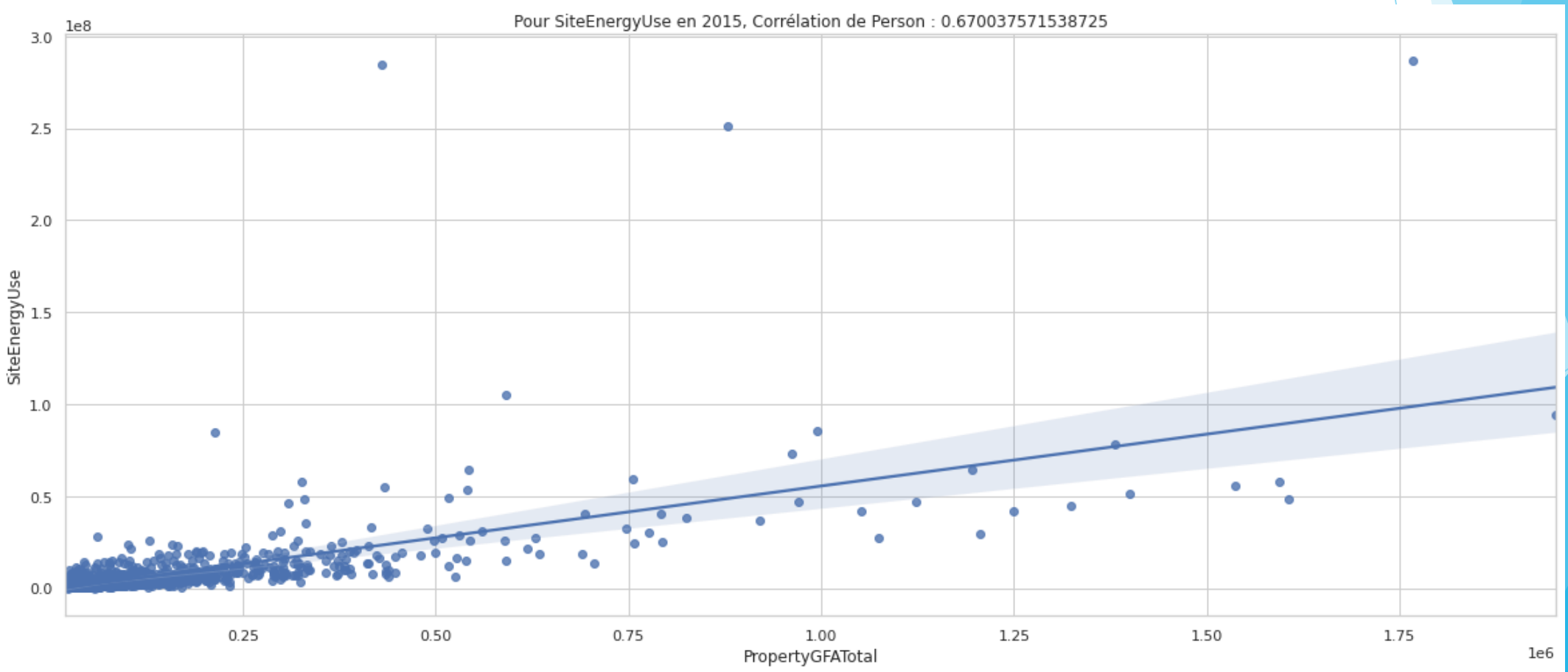
Relation entre 'TotalGHGEmissions' et 'PrimaryPropertyType'



ANOVA : $n^2 = 0.49$

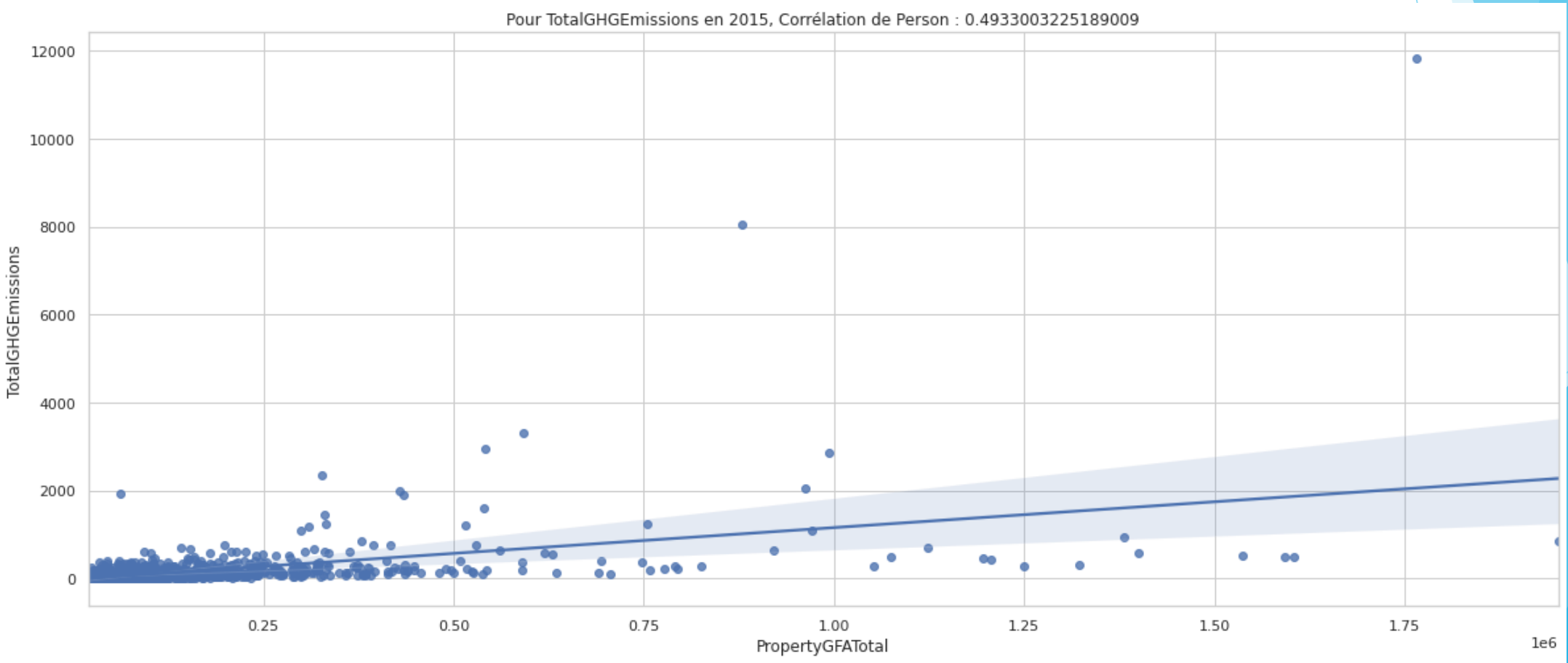
1. NETTOYAGE ET EXPLORATION

Relation entre 'SiteEnergyUse' et 'PropertyGFATotal'



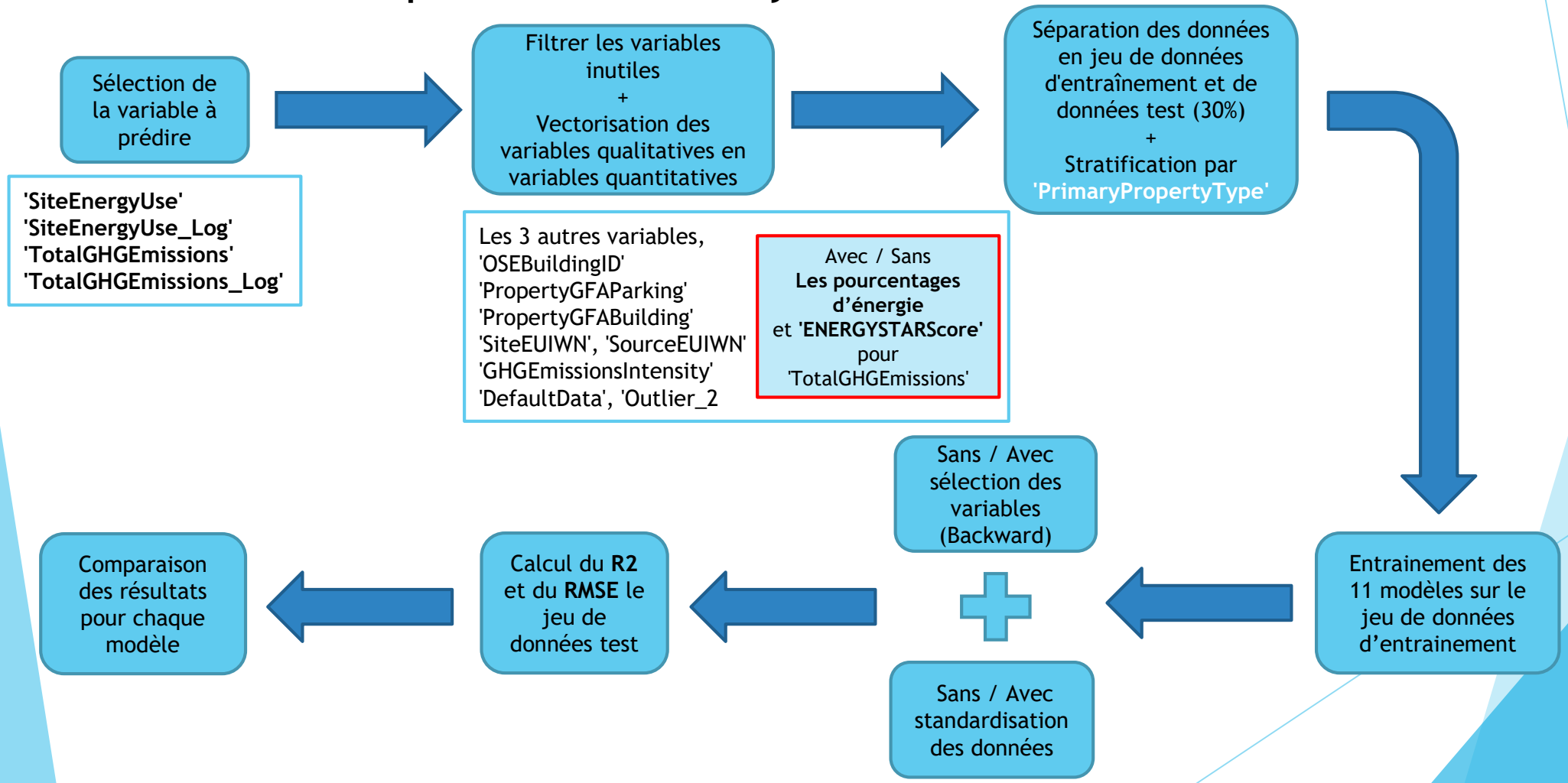
1. NETTOYAGE ET EXPLORATION

Relation entre 'TotalGHGEmissions' et 'PropertyGFATotal'



2. MODELISATION

Méthode utilisée pour réaliser les analyses sur les différents modèles :



2. MODELISATION

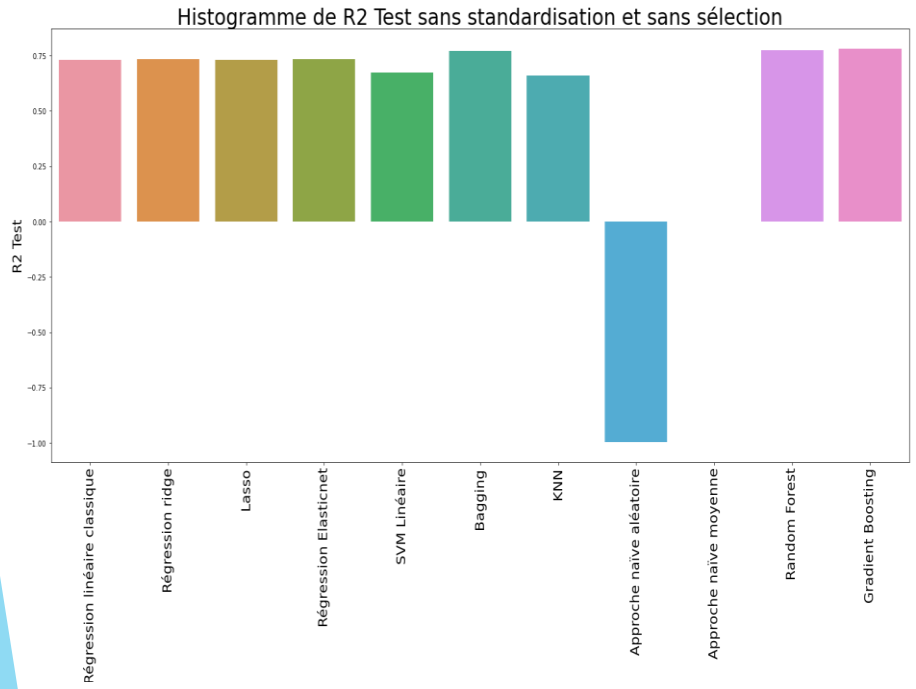
Liste des modèles testés :

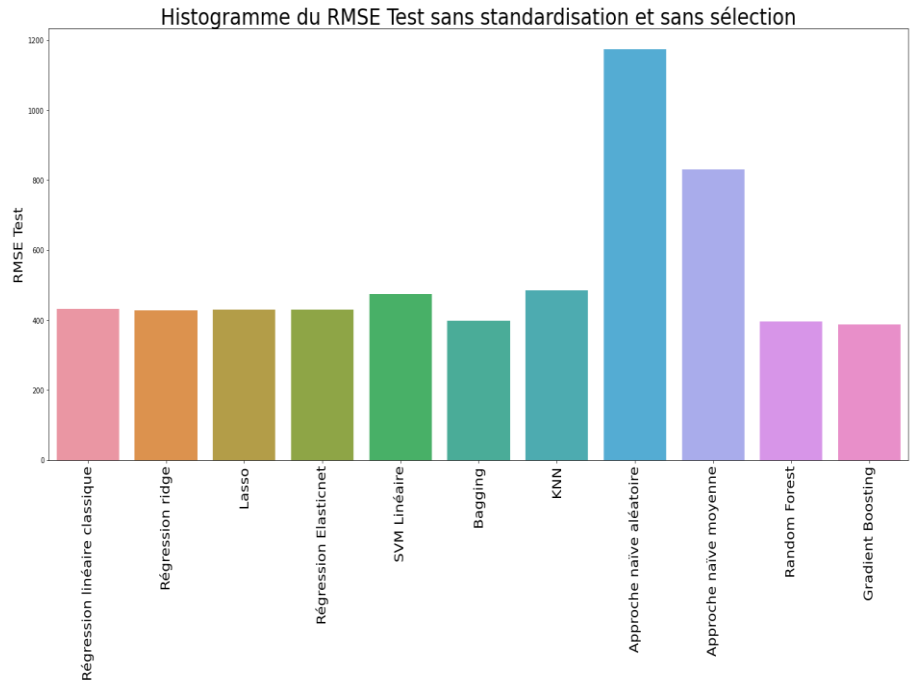
- Régression linéaire
- Approche naïve : prédire les valeurs aléatoirement
- Approche naïve : prédire les valeurs par ma moyenne
- Régression Ridge (alpha)
- Lasso (alpha)
- Régression Elasticnet (alpha, l1_ratio)
- SVM Linéaire (C)
- Bagging (n_estimators)
- K plus proches voisins (n_neighbors)
- Random Forest (n_estimators, max_depth, min_samples_split)
- Gradient Boosting (n_estimators, learning_rate)

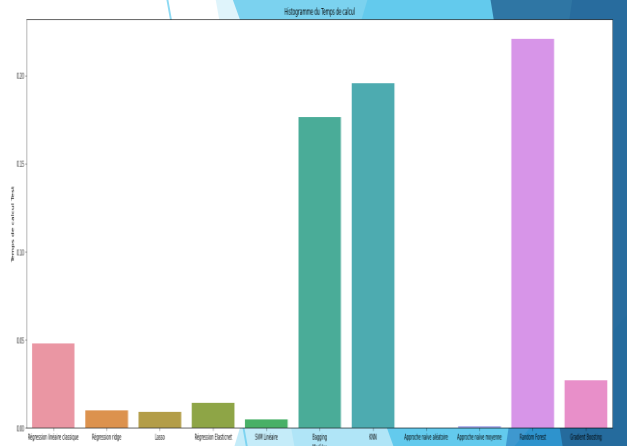
Recherche de l'hyperparamètre optimal par validation croisée avec GridSearchCV

3. Modèle optimal

Comparaison des résultats sur la variable 'SiteEnergyUse' sans les pourcentages d'énergie







Meilleurs résultats avec le Gradient Boosting avec R2 le plus grand et le RMSE le plus faible (résultats très proches avec/sans les pourcentages d'énergie et avec/sans transformation au Log)

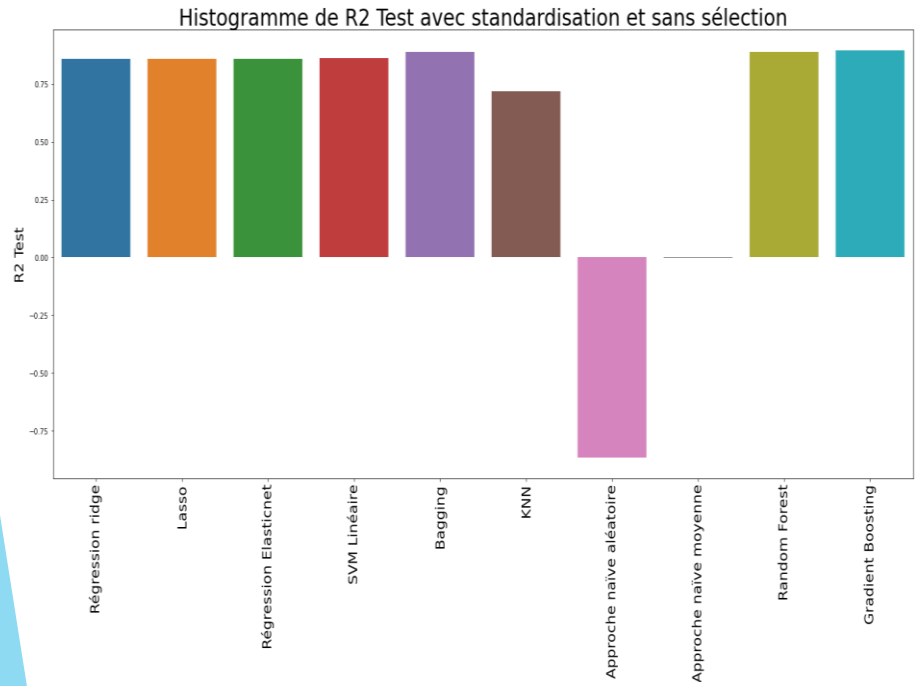
Sans les pourcentages : R2 = 0.78 RMSE = 387 Temps = 0.02 (sans standardisation)

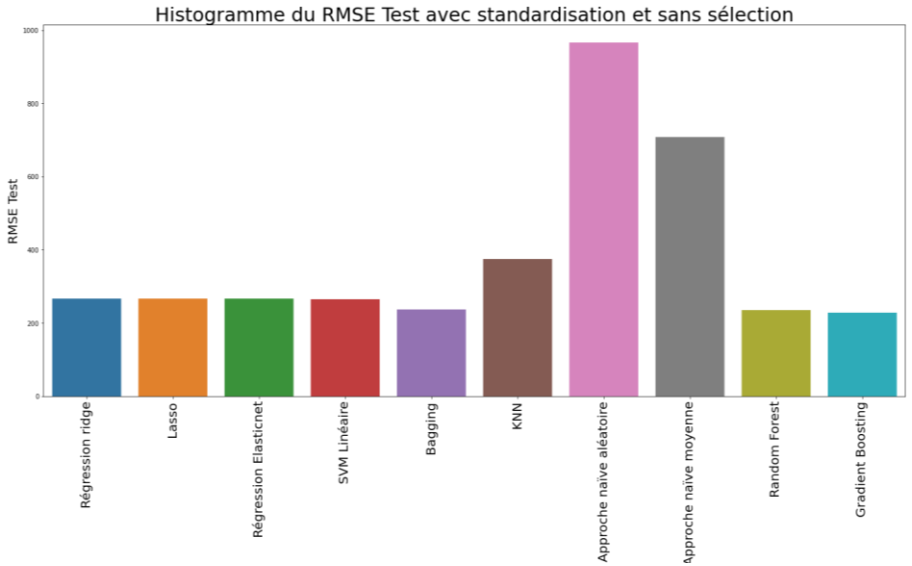
Avec les pourcentages : R2 = 0.80 RMSE = 365 Temps = 0.02 (avec standardisation)

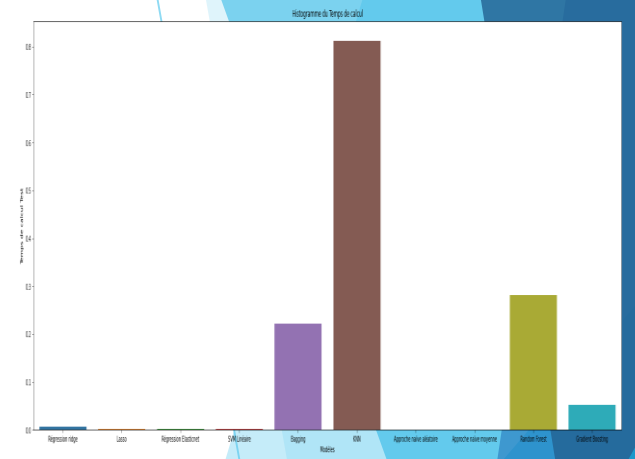
Avec 'SiteEnergyUse_Log' : R2 = 0.76 RMSE = 405 Temps = 0.03 (avec standardisation)

3. Modèle optimal

Comparaison des résultats sur la variable 'TotalGHGEmissions' avec les pourcentages d'énergie



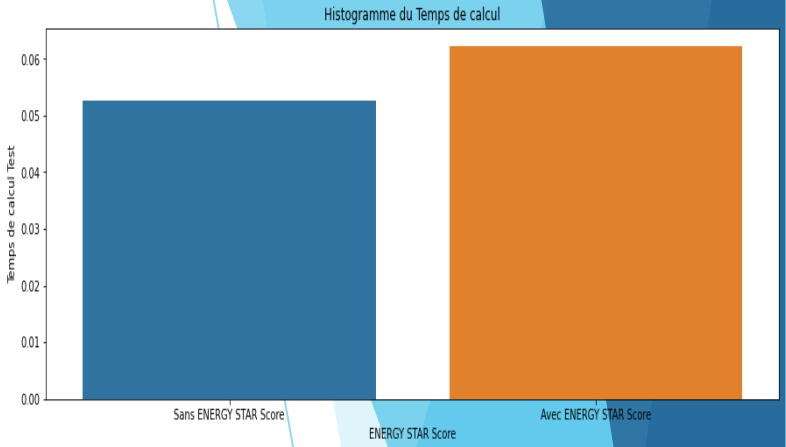




Meilleurs résultats avec le Gradient Boosting avec R2 le plus grand et le RMSE le plus faible (résultats très proches avec/sans transformation au Log mais différence des résultats en ajoutant les pourcentages d'énergie)

Sans les pourcentages :	R2 = 0.60	RMSE = 448	Temps = 0.01 (avec standardisation)
<u>Avec les pourcentages</u> :	R2 = 0.89	RMSE = 227	Temps = 0.05 (avec standardisation)
Avec 'TotalGHGEmissions_Log' :	R2 = 0.88	RMSE = 235	Temps = 0.05 (sans standardisation)

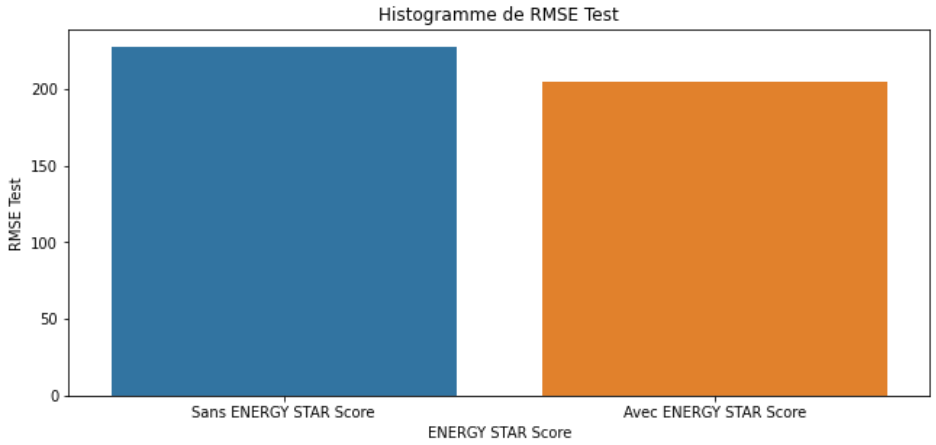
3. Modèle optimal

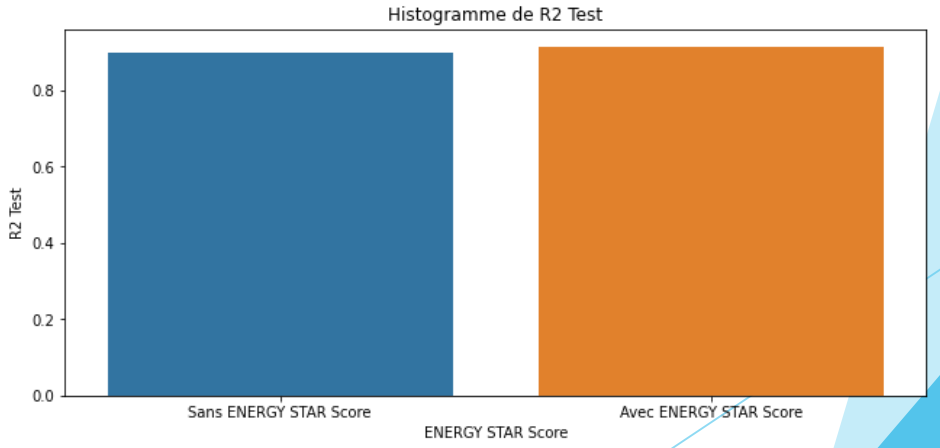


Intérêt de ENERGY STAR Score la variable 'TotalGHGEmissions' avec les pourcentages d'énergie

Modèle optimal = Gradient Boosting

- Avec ENERGY STAR Score, R2 = 0.91 et RMSE = 204 avec standardisation des données et sans sélection des variables
- Sans ENERGY STAR Score, R2 = 0.89 et RMSE = 227 avec standardisation des données et sans sélection des variables



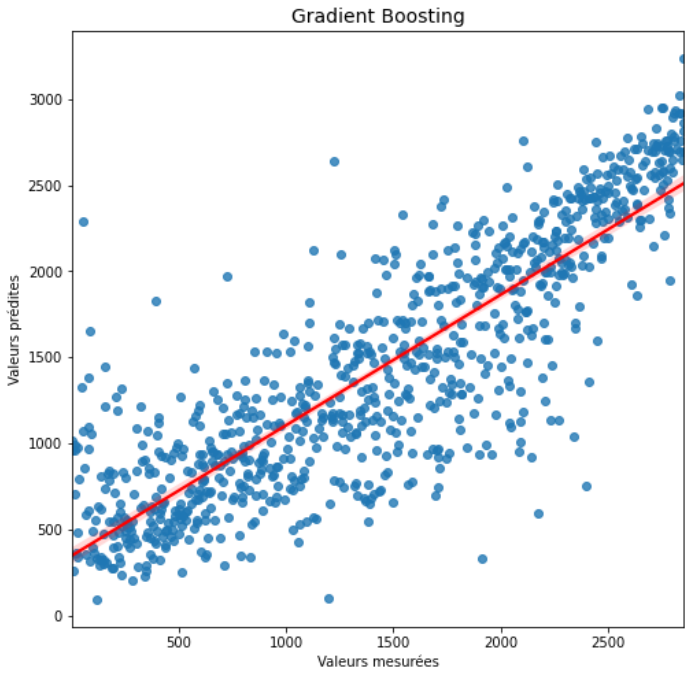


RMSE plus faible et R2 plus grand avec ENERGY STAR Score mais qui n'améliorent pas beaucoup le modèle avec des temps d'exécution proches

3. Modèle optimal

Modèle sélectionné = Gradient Boosting sans et avec les pourcentages d'énergie

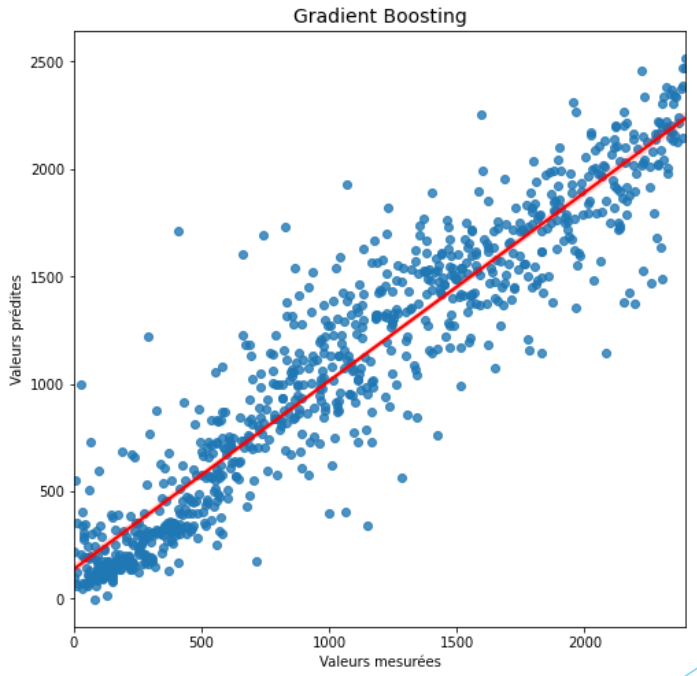
SiteEnergyUse



Sans standardisation des données
Sans sélection des variables

n_estimators = 750 R2 = 0.74
learning_rate = 0.03 RMSE = 419

TotalGHGEmissions



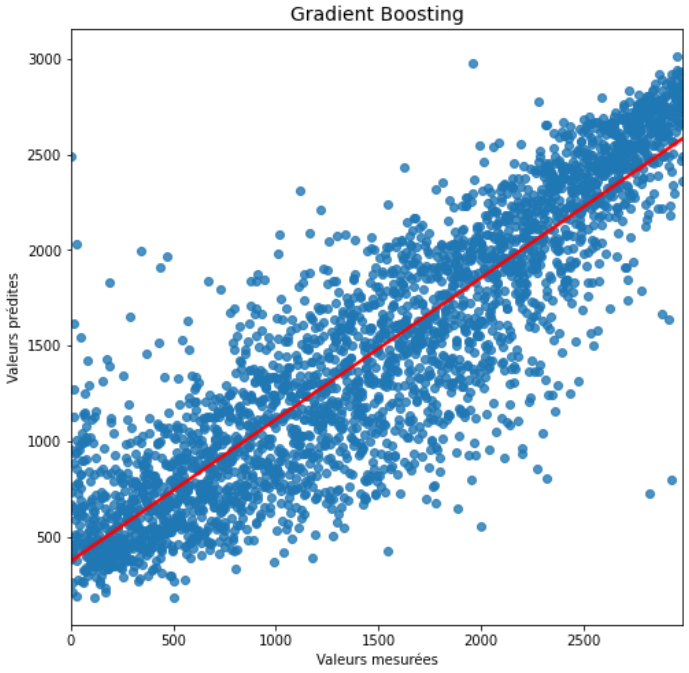
Avec standardisation des données
Sans sélection des variables
Sans ENERGY STAR Score

n_estimators = 850 R2 = 0.88
learning_rate = 0.03 RMSE = 241

3. Modèle optimal

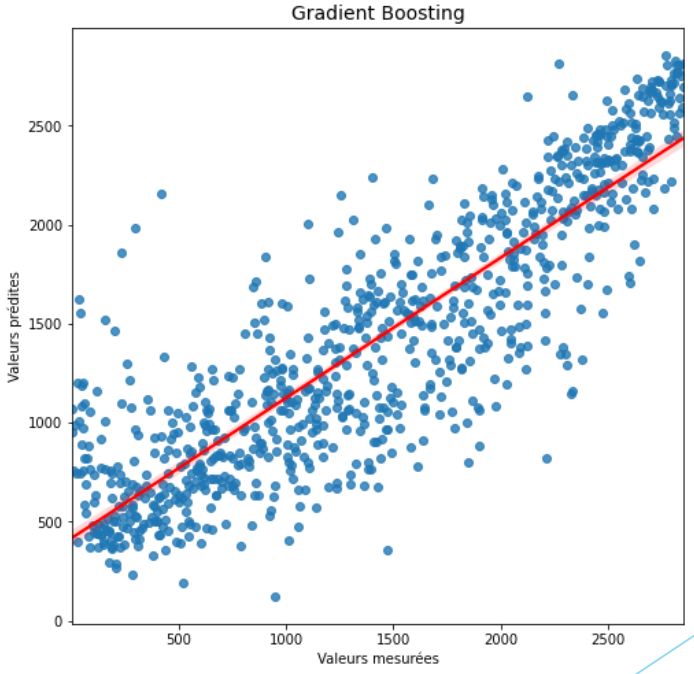
Modèle sur le DataFrame de 2016 avec le modèle de Gradient Boosting pour 'SiteEnergyUse'

Entrainé sur 2015



n_estimators = 550 R2 = 0.79
learning_rate = 0.03 RMSE = 396

Entrainé sur 2016

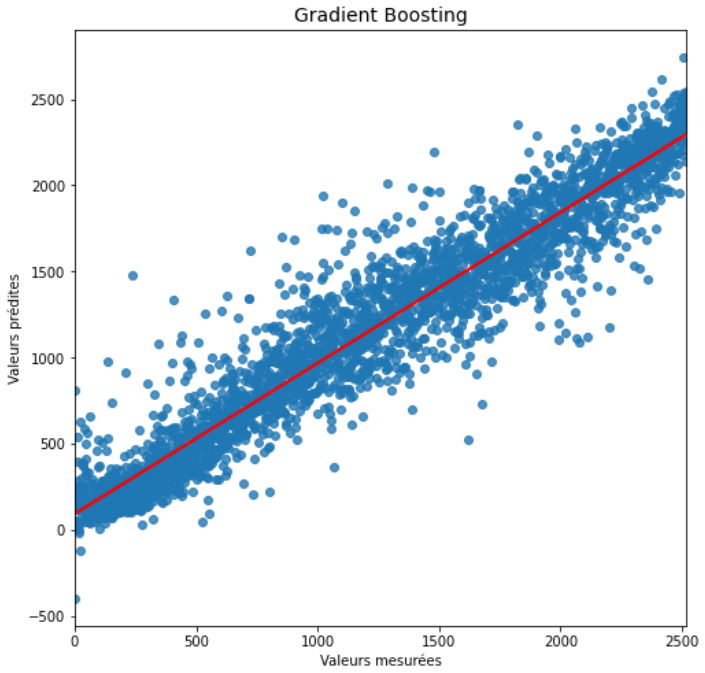


n_estimators = 350 R2 = 0.75
learning_rate = 0.03 RMSE = 412

3. Modèle optimal

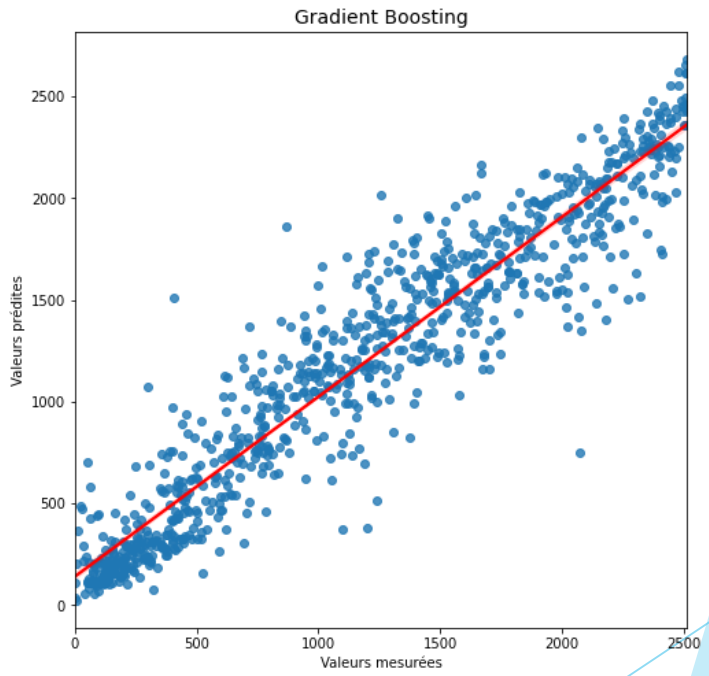
Modèle sur le DataFrame de 2016 avec le modèle de Gradient Boosting pour 'TotalGHGEmissions'

Entrainé sur 2015



n_estimators = 1550 R2 = 0.91
learning_rate = 0.03 RMSE = 221

Entrainé sur 2016



n_estimators = 550 R2 = 0.89
learning_rate = 0.04 RMSE = 236

4. Conclusion

Réponse à la problématique : Modèle optimal pour prédire les **émissions de CO2** et la **consommation totale d'énergie** = Gradient Boosting

Pas nécessaire d'utiliser l'"ENERGY STAR Score" pour prédire les émissions de CO2

Intérêt d'ajouter les pourcentages des énergies dans le modèle :

- Si données récupérables ou estimables
- Améliorent le modèle (plus performant et plus rapide surtout pour 'TotalGHGEmissions')

Meilleurs résultats en entrainant sur le jeu de données de 2015 et en testant sur le jeu de données de 2016 :

- Jeux de données plus grands
- Teste sur les mêmes bâtiments
- Pas d'effet de l'année

MERCI DE VOTRE ATTENTION

QUESTIONS – REPONSES