

# **Anticipation de la consommation d'énergie et d'émission de CO<sub>2</sub>**



Mitra DADGAR-Data Scientist

# Contenu

- 1 INTRODUCTION
- 2 NETTOYAGE ET ANALYSE DESCRIPTIVE
- 3 MACHINE LEARNING ET MODELISATION
- 4 CONCLUSION ET AMÉLIORATIONS

# Introduction



**Objectifs:** Être une ville neutre en émissions de gaz à effets de serre pour 2050

- Données : Deux jeux de données sur les bâtiments non-résidentiels (2015 et 2016)
- Prédire les émissions de CO2
- Prédire la consommation d'énergie
- On cherche également à évaluer l'intérêt de l'"ENERGY STAR Score" pour la prédiction d'émission



# DataSet

Deux jeux de données sur les bâtiments non-résidentiels (2015 et 2016) , des données liées aux informations :

- ❖ Usage de la propriété
- ❖ Date de construction
- ❖ Nombre de bâtiments et d'étages
- ❖ Superficie de la propriété (bâtiments, parking et autres)
- ❖ Localisation (quartier, adresse et géolocalisation)

Rassemblement des données de 2015 et 2016 afin de fusionner mieux les deux datasets:

- ❖ La colonne Location en dataset 2015 contient les colonnes latitude, longitude, address, city, state et zip de dataset 2016 donc on va séparer cette colonnes à 6 colonnes
- ❖ Harmonisation des chaînes de caractères dans les données de 2015

# **Nettoyage et Analyse descriptive**

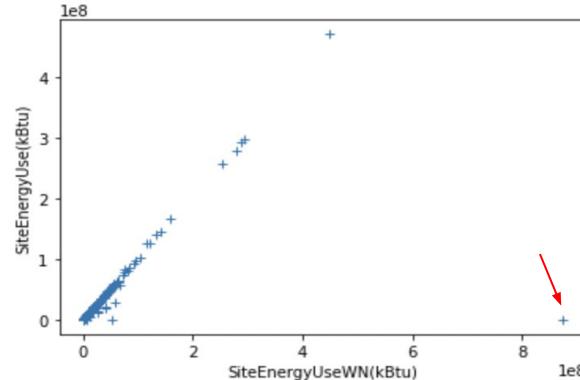
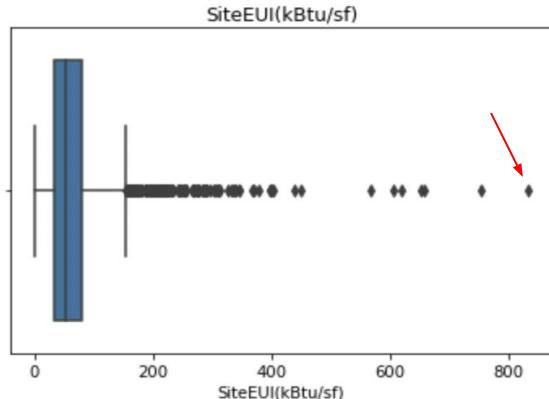


# Data cleaning

- ❖ Suppression des colonnes avec plus de 75% de valeurs manquantes
- ❖ Suppression des bâtiments résidentiels
- ❖ Suppression des données doublons (1620 valeurs doublantes pour numéro OSEBuildingID et l'Address)
- ❖ Suppression les colonnes non utilisées

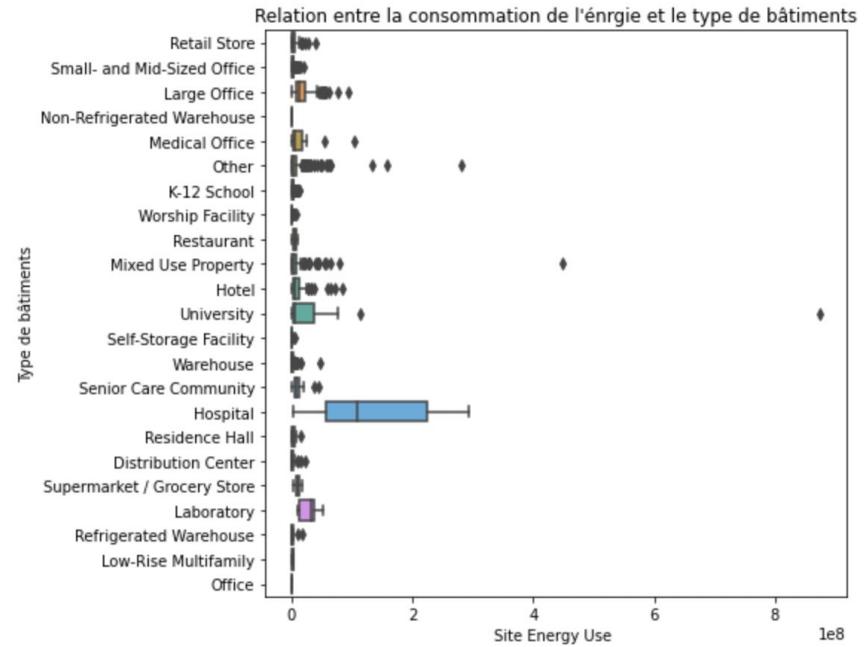
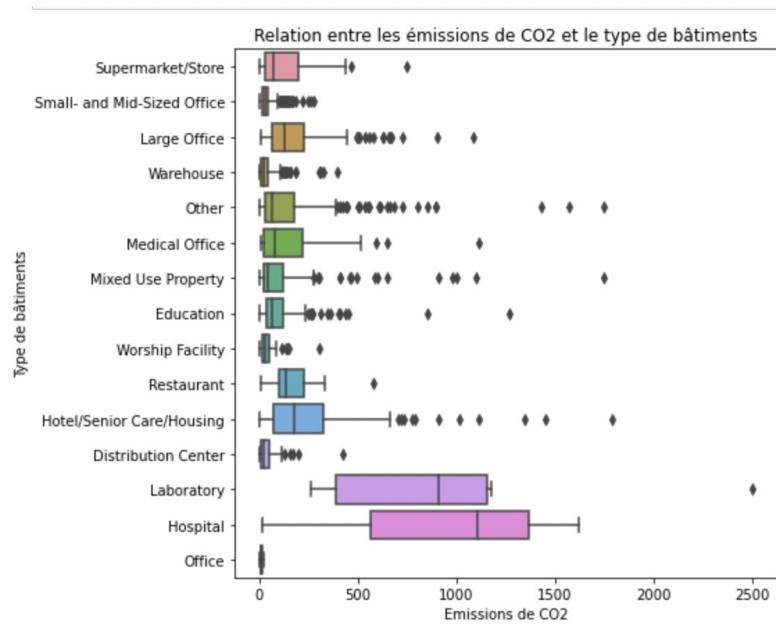
('Electricity(kWh)', 'NaturalGas(therms)', 'OtherFuelUse(kBtu)', 'GHGEmissionsIntensity', 'DefaultData', 'ComplianceStatus', 'SourceEUIWN(kBtu/sf)', 'SiteEUIWN(kBtu/sf)',  
'SiteEnergyUseWN(kBtu)', 'Seattle Police Department Micro Community Policing Plan Areas', 'SPD Beats')

- ❖ Retrait des valeurs aberrantes (On supprime toutes les valeurs énergétiques qui sont inférieures à 0 )
- ❖ Supprimer Outliers SiteEnergyUse quantile plus (0.98)



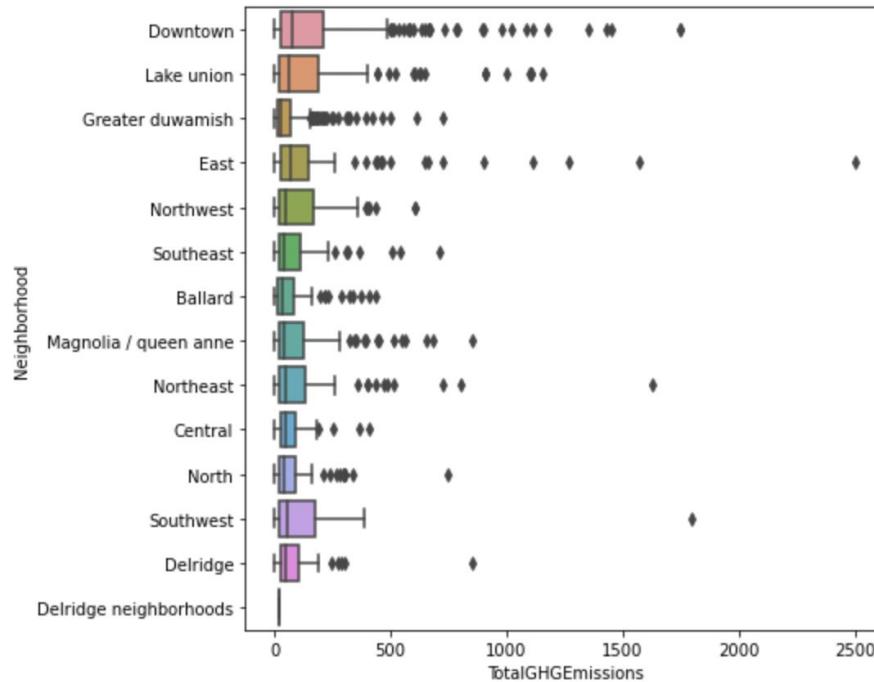
# Data exploration

Si le type de bâtiment a un effet sur la consommation d'énergie et l'émission de CO2?



Les hôpitaux et les laboratoires sont les bâtiments les plus polluants

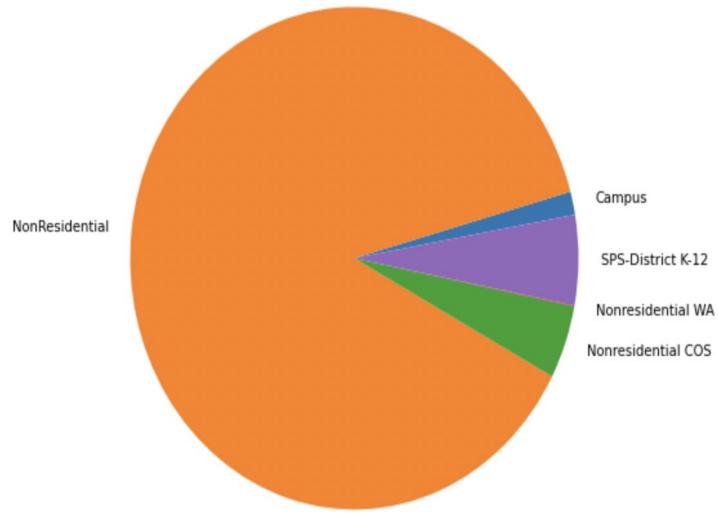
# Influence du quartier sur la Emissions de CO2



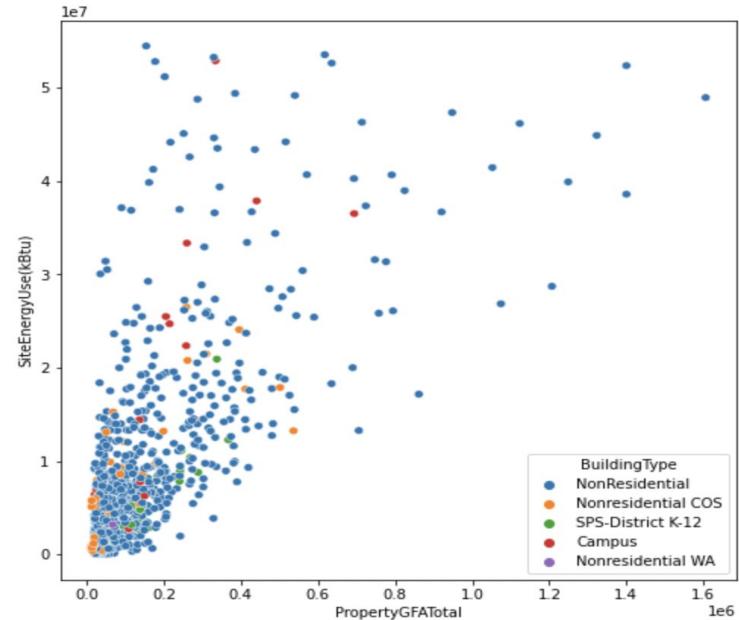
# Répartition des divers types de bâtiments

## Distribution des consommations par surface et type de bâtiment

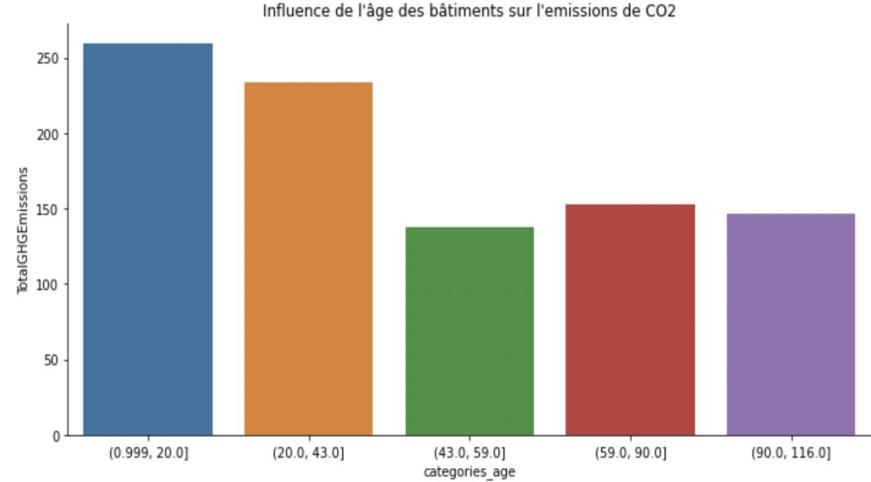
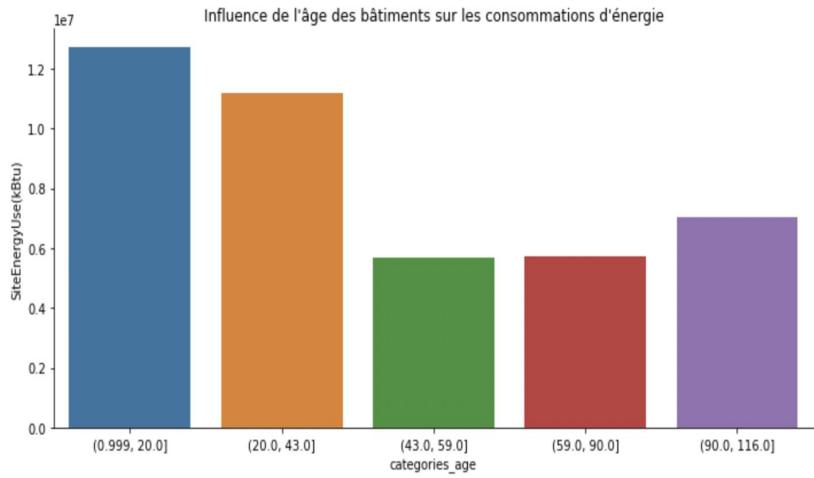
Répartition des types de bâtiments du Dataset



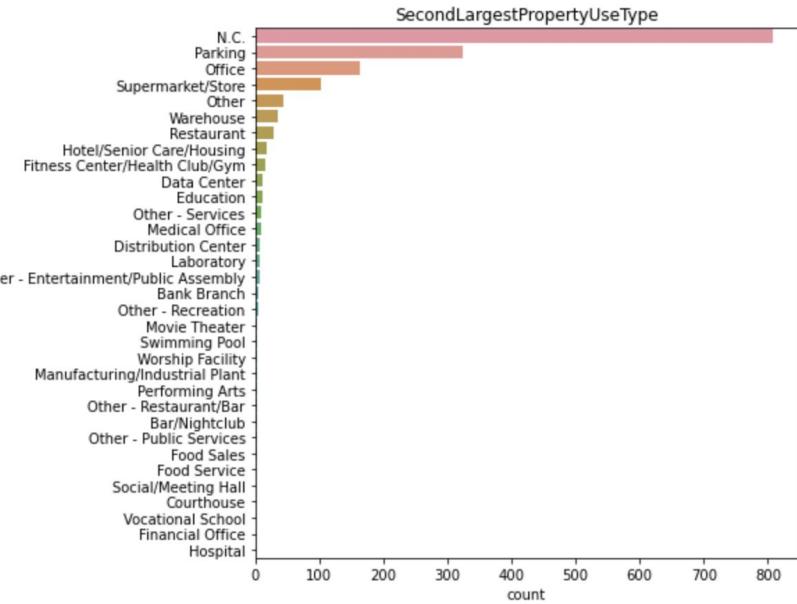
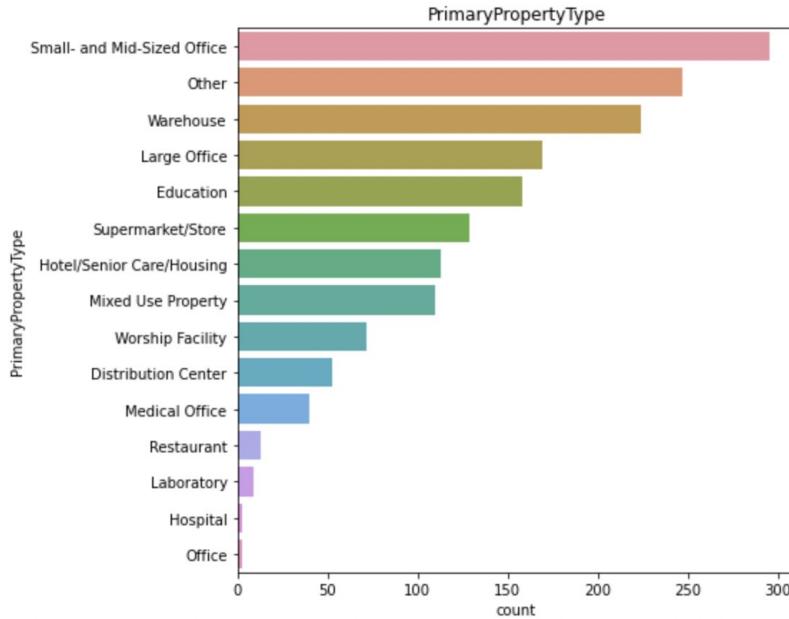
Consommations d'énergie par surface totale au sol et par type de bâtiment



# Influence de l'âge sur les targets

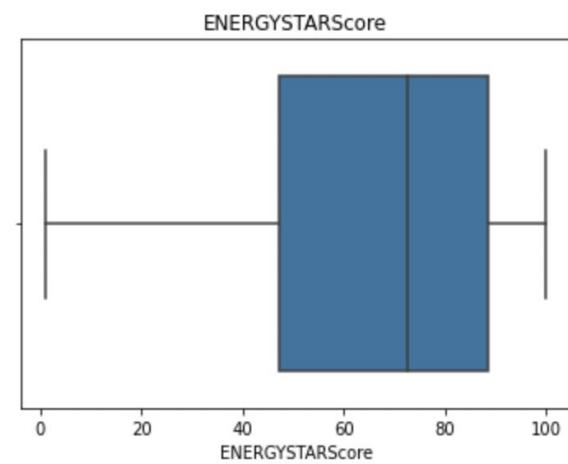
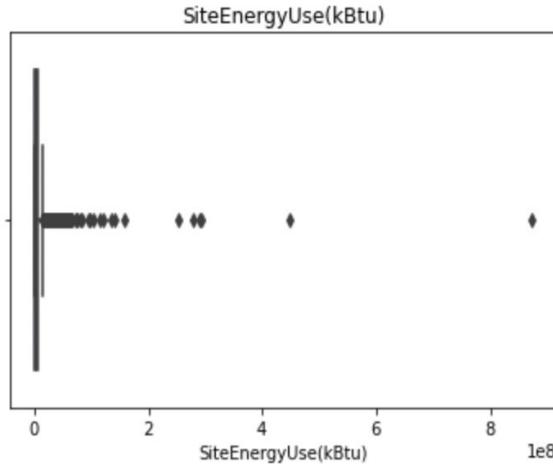
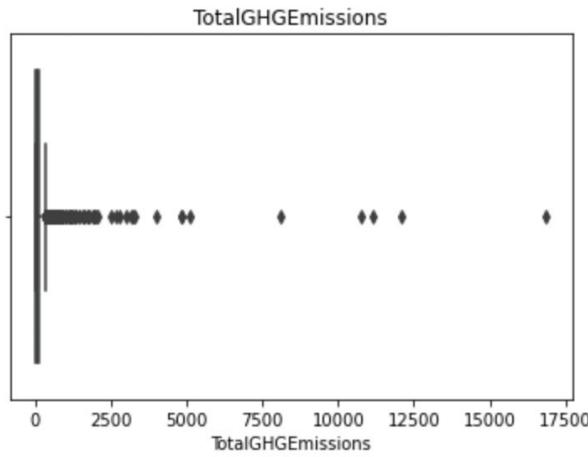


# Distribution des types de bâtiments



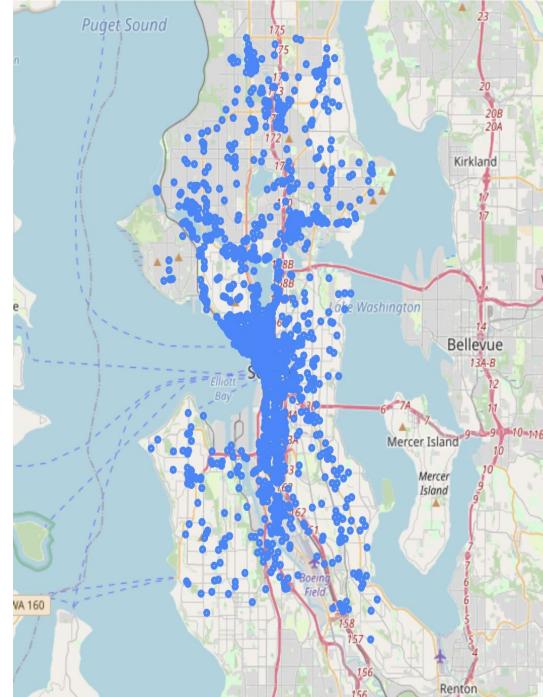
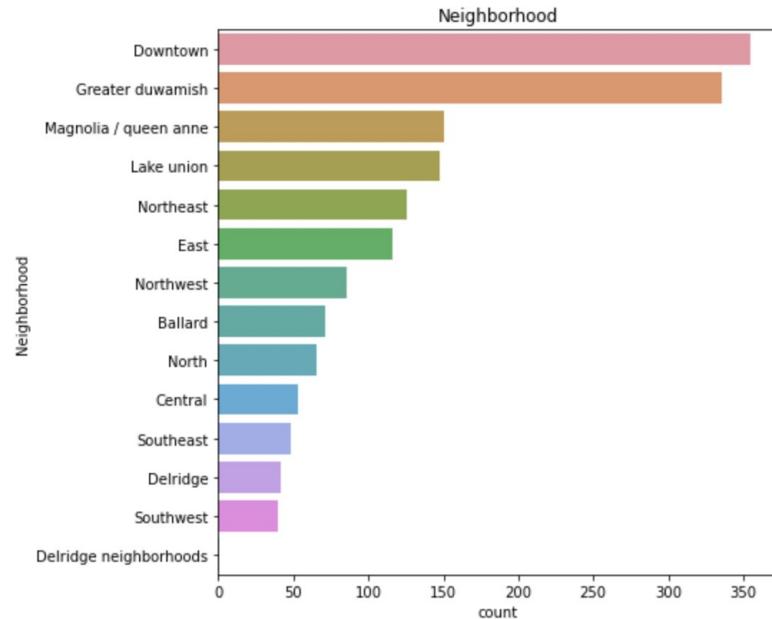
# Distribution des mesures énergétiques

Boîte à moustache de la variable « Total GHG Emissions , Site Energy Use, Energy Star Score »



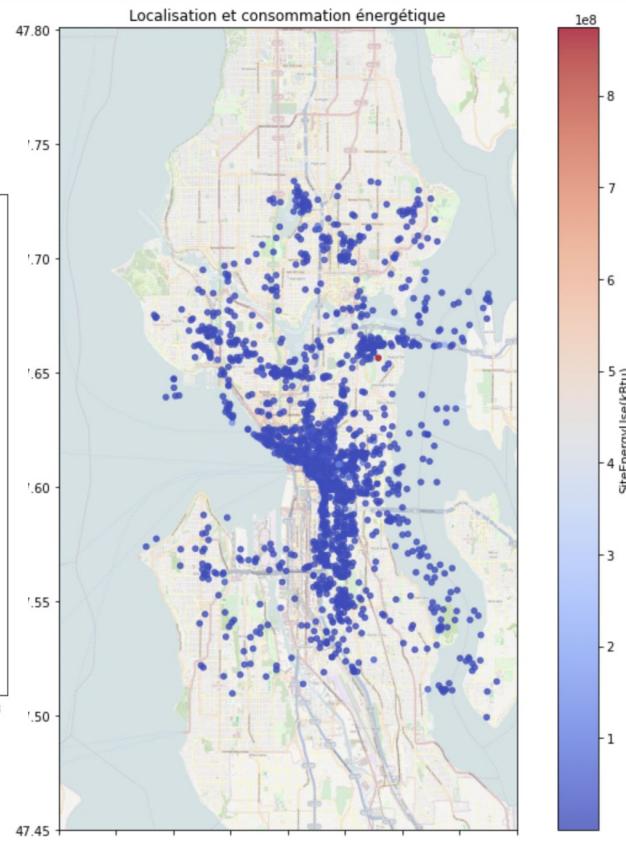
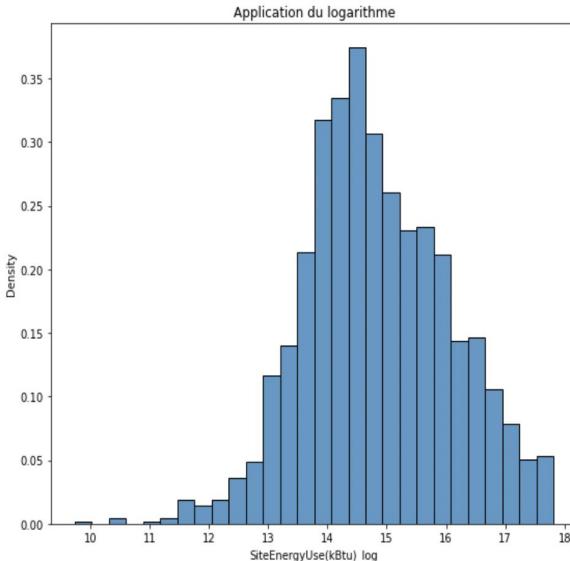
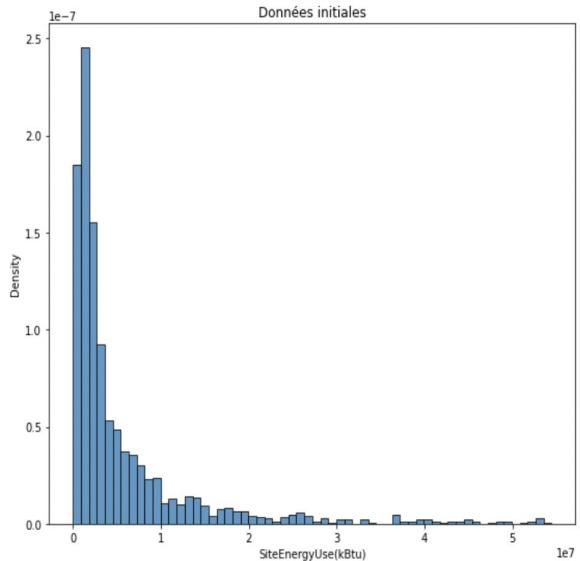
# Répartition des bâtiments en distribution Quartier

La répartition des localisation des bâtiments dans Seattle



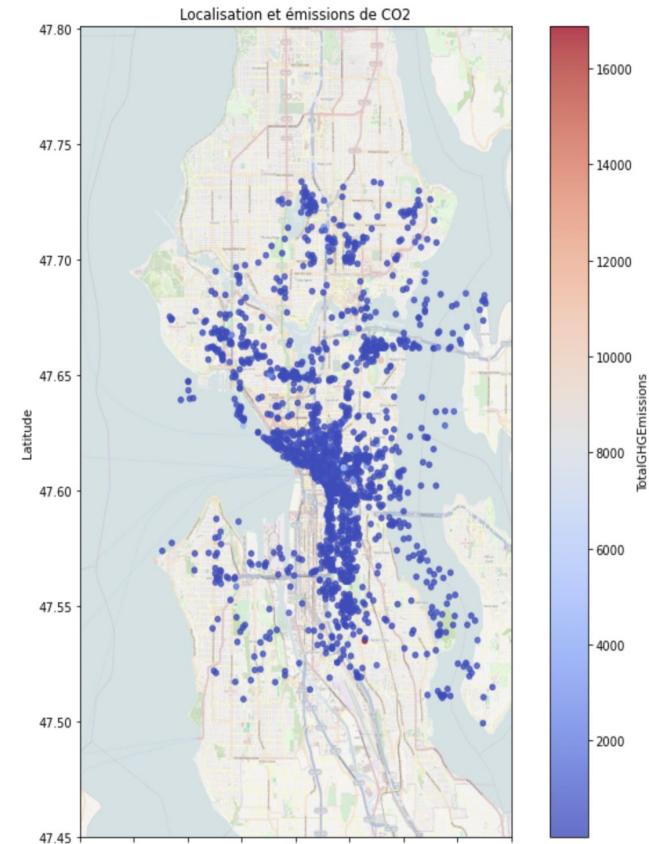
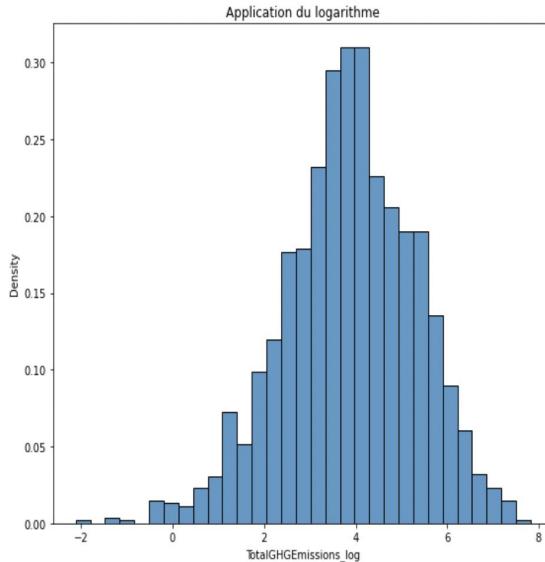
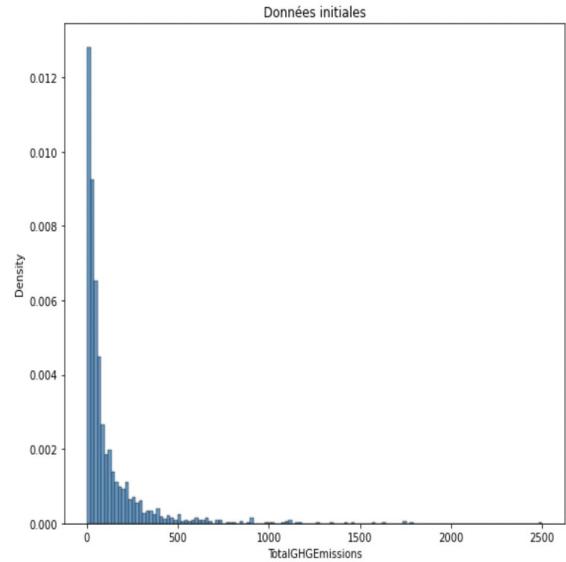
# Analyse de SiteEnergyUse

Distribution des consomations energy avec changement d'échelle logarithmique



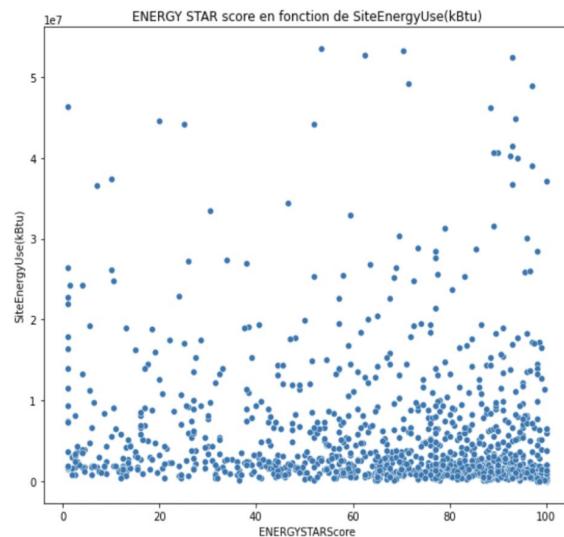
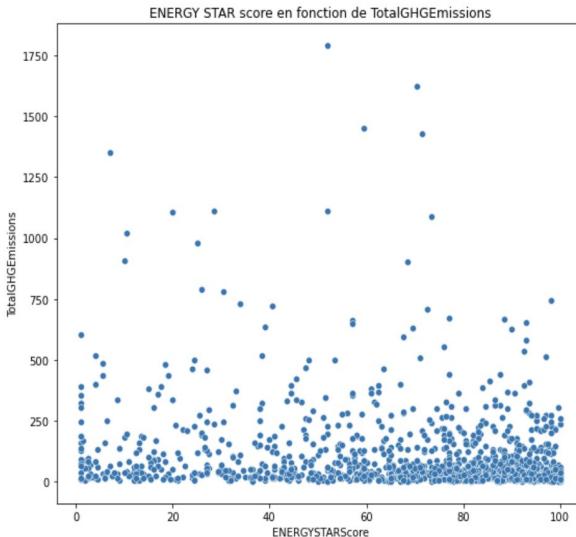
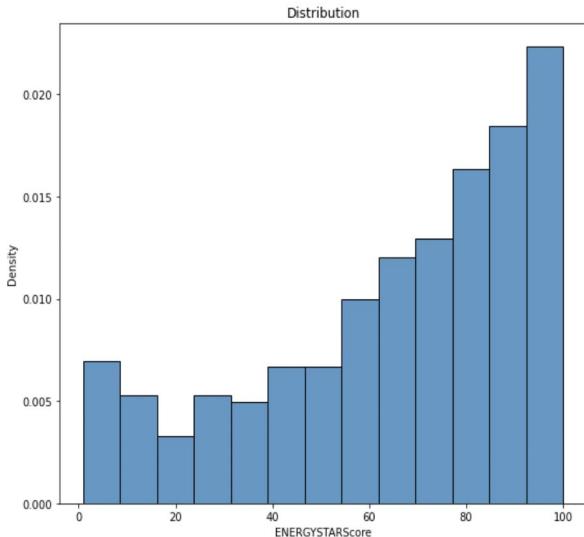
# Analyse de TotalGHGEmissions

Distribution des émissions de CO<sub>2</sub> avec changement d'échelle logarithmique



# Distribution de EnergyStarScore ainsi que sa relation avec les émissions de CO2 et consommation d'énergie

Analyse de la variable ENERGY STAR Score



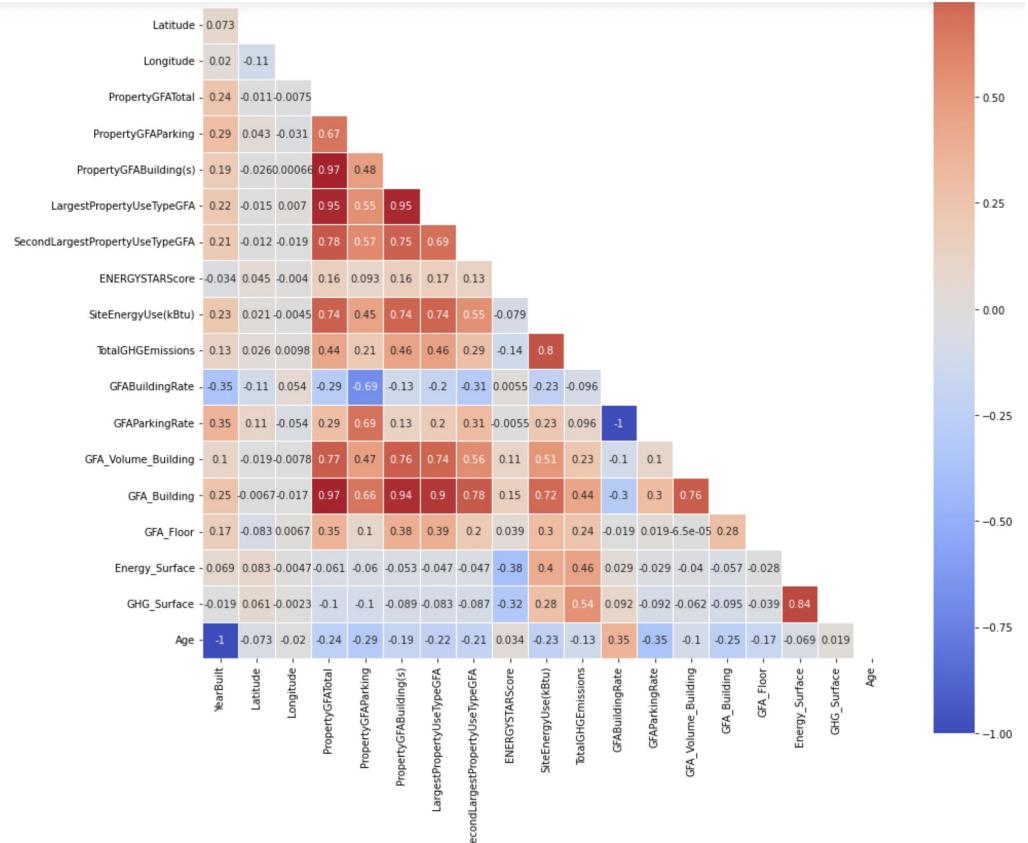
La distribution ne suit pas de loi normale et la majorité des bâtiments à un score supérieur à 50. Il semble que le score ENERGY STAR n'a pas une corrélation importante avec les émissions de CO2 et la consommation d'énergie.

# Corrélation linéaire des variables

Corrélation forte entre les variables cibles 0.84

Prédire l'une des variables target aide à prédire l'autre

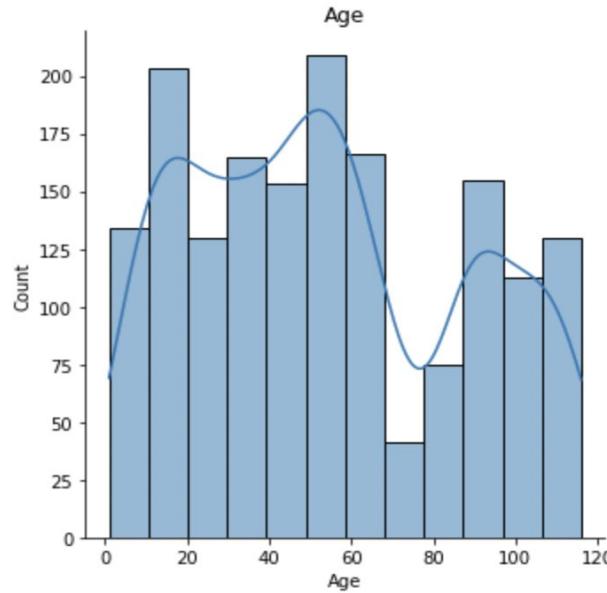
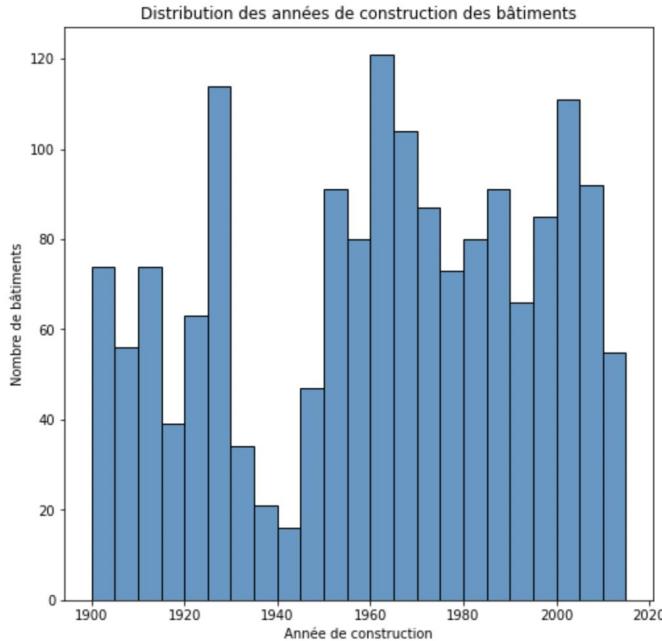
PropertyGFATotal et LargestPropertyUseTypeGFA sont liés à TotalGHGEmissions et également à SiteEnergyUse



# Feature engineering

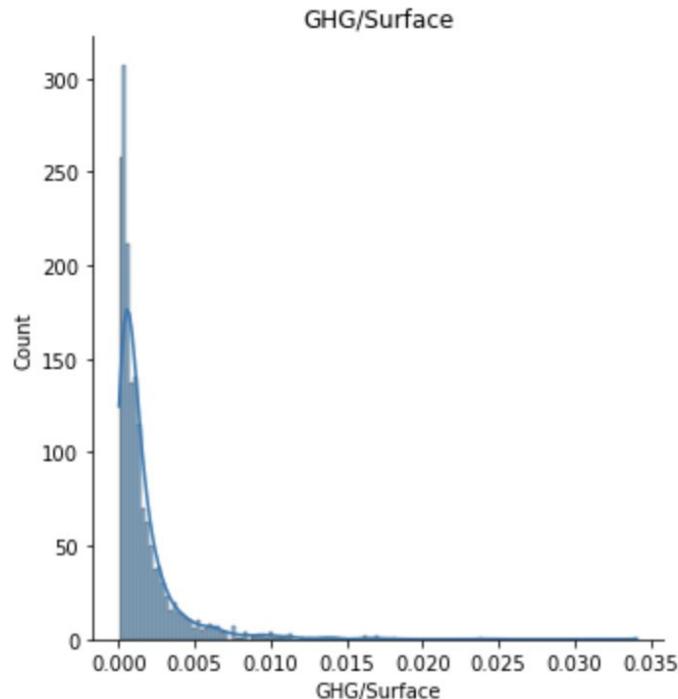
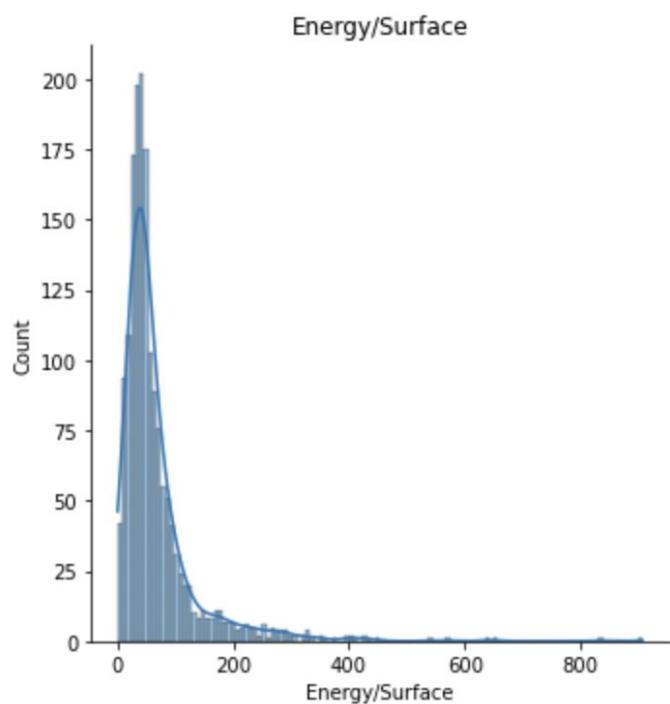
Création de nouvelles variables pour l'exploration :

- ❖ (data leakage) Energy/Surface : la consommation d'énergie sur surface (`SiteEnergyUse(kBtu)` / `PropertyGFATotal`)
- ❖ (data leakage) GHG/Surface: l'émissions de CO<sub>2</sub> GHG sur surface (`TotalGHGEmissions` / `PropertyGFATotal`)
- ❖ Âge de bâtiments (Année – Année de construction)



# Feature engineering

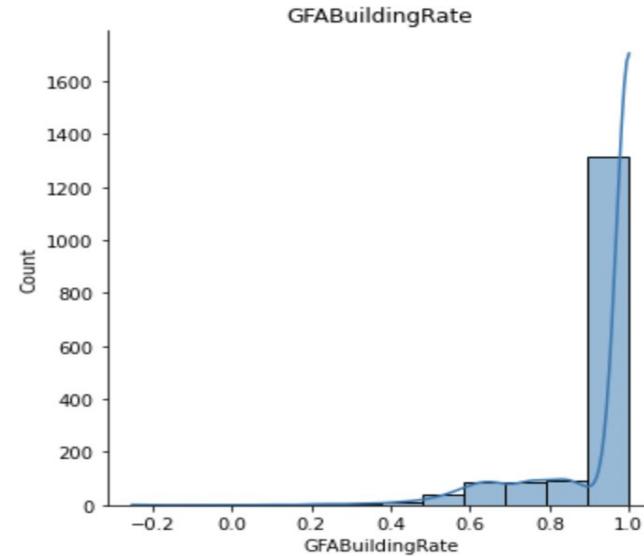
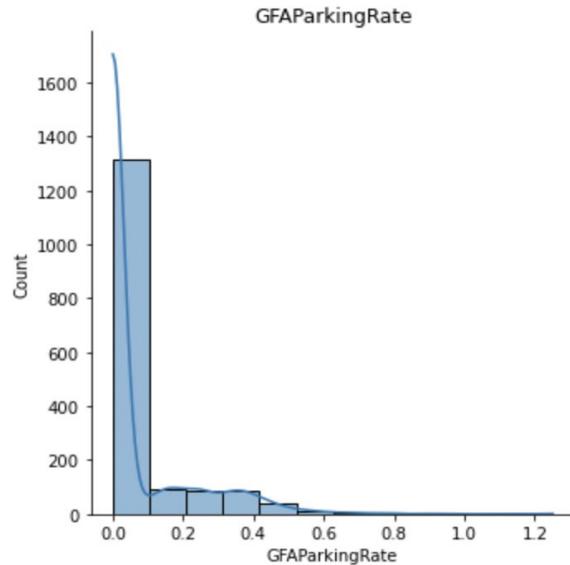
Distribution la consommation d'énergie sur surface et l'émission de CO<sub>2</sub> GHG sur surface.



# Feature Engineering

Création de nouvelles variables:

- ❖ surface bâtiment par rapport surface parking



# Machine Learning et Modelisation

Un modèle simple de **régression linéaire multivariée** ne pourrait pas répondre totalement à notre problème de prédiction. Nous allons donc utiliser ce premier modèle comme baseline et tester les métriques principales :

- ❖ MAE
- ❖ MSE
- ❖ RMSE
- ❖ R<sup>2</sup>
- ❖ Median abs err



# Encoding et standardisation

- Supprimer les colonnes supplémentaires (redondance, data leakage, identifiants)

[ 'OSEBuildingID', 'DataYear', 'PropertyName', 'TaxParcelIdentificationNumber', 'CouncilDistrictCode', 'ZipCode', 'Address', 'City', 'State', 'NumberofBuildings', 'NumberofFloors',  
'SiteEUI(kBtu/sf)', 'SourceEUI(kBtu/sf)', 'SteamUse(kBtu)', 'Electricity(kBtu)', 'NaturalGas(kBtu)', ]

- Séparation des variables catégorielles et numériques:

- Variables catégorielles : Imputer → Mode - OneHotEncoding
- Variables numériques : Imputer → Médiane - StandardScaler

---

```
[ 'BuildingType', 'PrimaryPropertyType', 'Neighborhood', 'ListOfAllPropertyUseTypes', 'LargestPropertyUseType', 'SecondLargestPropertyUseType' ]
```

---

```
=====
```

```
[ 'Latitude', 'Longitude', 'PropertyGFATotal', 'LargestPropertyUseTypeGFA', 'SecondLargestPropertyUseTypeGFA', 'TotalGHGEmissions', 'GFABuildingRate', 'GFAParkingRate', 'Age' ]
```

---

# Cross validation et métriques

Evaluation des performances du modèle :

- MAE (Erreur absolue moyenne)
- MSE (Erreur quadratique moyenne)
- RMSE (La racine de l'erreur quadratique moyenne)
- R<sup>2</sup> (Coefficient de détermination ou R carré)

Recherche des hyperparamètres optimaux :

- Grille de recherche pour explorer toutes les combinaisons
- Validation croisée pour estimer la performance de la combinaison
- GridSearchCV outil combinant recherche par grille et CV

# Présentation des modèles utilisés- TotalGHGEmissions

Tests de plusieurs modèles de machine learning sans fixer des paramètres:

- Linéaire avec pénalisation (Régression linéaire, Lasso, Ridge, SVM)
- Non-linéaire (Ridge Kernel, decision tree) et ensembliste (Random Forest, XGBoost)

model	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
Dummy Regressor	84.421	28298.535	168.222	-0.118	35.665
Ridge	74.961	17137.618	130.911	0.323	36.768
Lasso	75.601	16882.240	129.932	0.333	43.910
DecisionTree	82.152	24946.712	157.945	0.014	31.685
ElasticNet	80.660	18775.042	137.022	0.258	52.656
SVR	77.520	25535.770	159.799	-0.009	31.391
Ridge Kernel	74.995	17175.507	131.055	0.321	35.881
RandomForestRegressor	65.329	15120.372	122.965	0.403	28.040
XGBRegressor	68.960	18122.994	134.622	0.284	28.105

Évaluer la performance Linear Regression...

MAE = 86718754349495.77  
MSE = 1.5035021320468986e+29  
RMSE = 387750194332240.0  
R<sup>2</sup> = -5.93979586732406e+24  
median abs err = 44.98

- Les modèles non linéaires le modèle de Random Forest et l'algorithme de XGBoost sont ceux qui obtiennent les meilleures performances

# Performances des modèles - SiteEnergyUse

Tests de plusieurs modèles de machine learning sans fixer des paramètres:

model	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
Dummy Regressor	3745887.344	4.747718e+13	6890368.831	-0.130	1561839.344
Ridge	2211600.620	1.326712e+13	3642406.310	0.684	1211141.366
Lasso	2372713.774	1.541869e+13	3926663.576	0.633	1411542.564
DecisionTree	2462189.335	1.873190e+13	4328036.453	0.554	1096343.625
ElasticNet	2610599.245	1.627664e+13	4034431.828	0.613	1795058.649
SVR	3745870.501	4.747682e+13	6890342.674	-0.130	1561853.529
Ridge Kernel	2217364.313	1.333549e+13	3651778.502	0.683	1195314.239
RandomForestRegressor	1872395.903	1.075487e+13	3279461.517	0.744	800027.973
XGBRegressor	1862489.747	1.010788e+13	3179289.566	0.759	861263.500

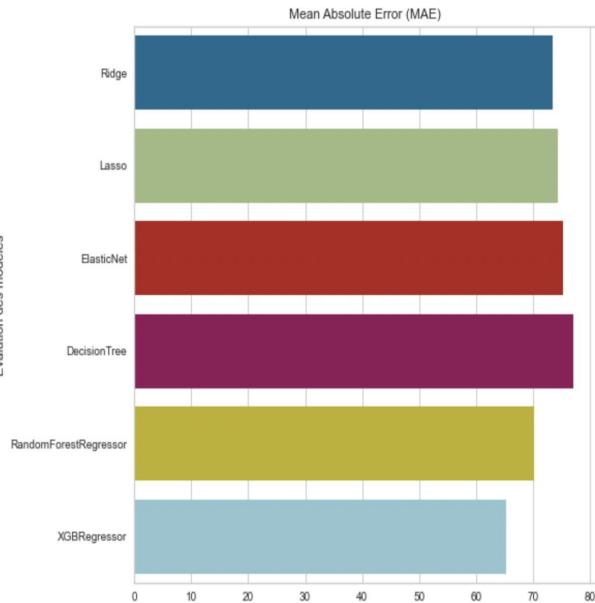
# Hyperparamètres TotalGHGEmissions

Tests de plusieurs modèles de machine learning avec fixer des paramètres:

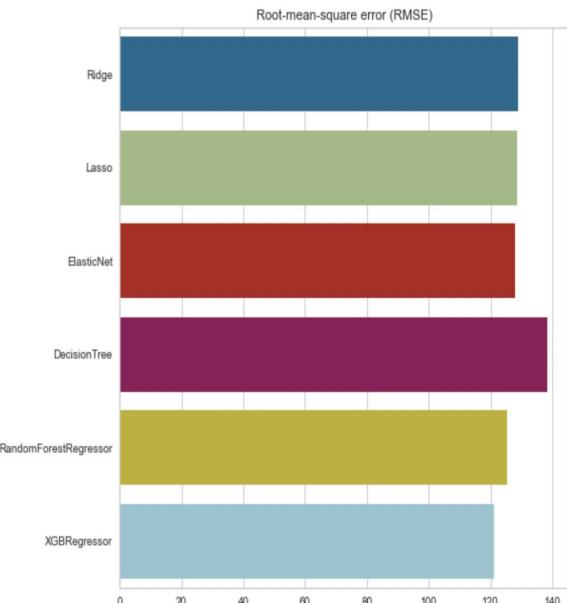
model	best_score	best_params	mean_test_score	mean_score_time	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
Ridge	0.398844	{'ridge_alpha': 3.593813663804626, 'ridge_co...	[0.24646927745244174, 0.24716007658165395, 0.2...	[0.006150674819946289, 0.005318260192871094, 0...	73.435	16621.990	128.926	0.343	35.466
Lasso	0.358209	{'lasso_alpha': 0.2782559402207126}	[0.24156085089199805, 0.24163154425765718, 0.2...	[0.005522727966308594, 0.0047207832336425785, ...	74.452	16535.880	128.592	0.347	39.540
ElasticNet	0.320966	{'elasticnet_alpha': 0.021544346900318846, 'e...	[0.17941622476346225, 0.17941622476346225, 0.1...	[0.007697343826293945, 0.005574703216552734, 0...	75.240	16376.136	127.969	0.353	38.543
DecisionTree	0.186262	{'decisiontreeregressor_max_depth': 3, 'decis...	[0.03497041783250412, 0.03497041783250412, 0.0...	[0.0058135509490966795, 0.0041691303253173825,...	77.158	19166.747	138.444	0.243	46.851
RandomForestRegressor	0.360241	{'randomforestrgressor_bootstrap': True, 'ra...	[0.06490053136937587, 0.06622847895968616, 0.0...	[0.008896827697753906, 0.01146559715270996, 0....	70.237	15671.695	125.187	0.381	39.021
XGBRegressor	0.459060	{'xgbregressor_learning_rate': 0.05, 'xgbregr...	[0.40434277792838935, 0.42741611449099537, 0.4...	[0.010956239700317384, 0.010713958740234375, 0...	65.217	14629.623	120.953	0.422	29.443

# Comparer les modèles \_ TotalGHGEmissions

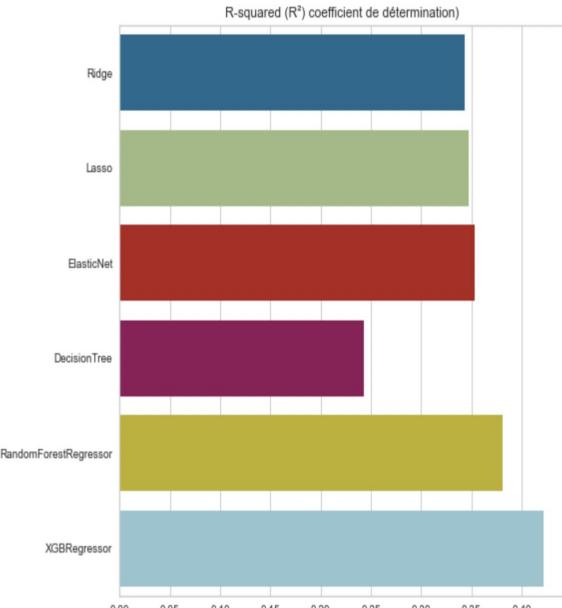
Evaluation des modèles



Evaluation des modèles



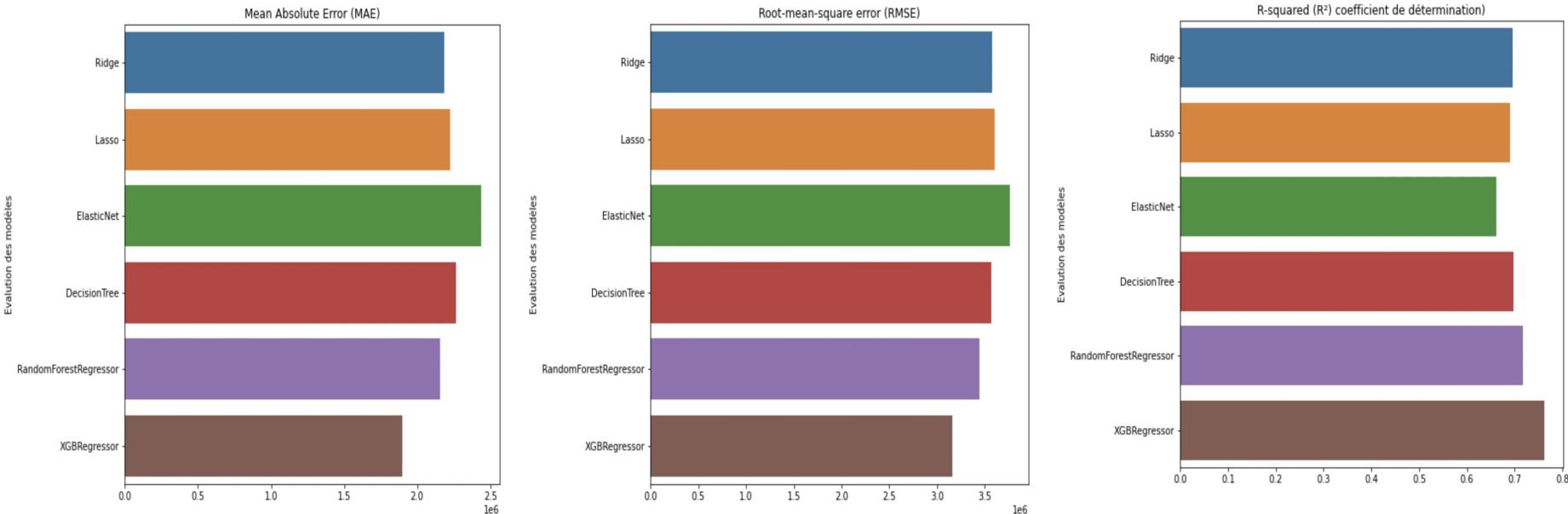
Evaluation des modèles



# Hyperparamètres SiteEnergyUse

model	best_score	best_params	mean_test_score	mean_score_time	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
Ridge	0.649723	{'ridge_alpha': 3.593813663804626, 'ridge_co...	[0.5629102946753006, 0.5635497386871807, 0.562...	[0.009410381317138672, 0.010222721099853515, 0...	2185841.789	1.276844e+13	3573295.912	0.696	1194681.702
Lasso	0.612418	{'lasso_alpha': 7742.636826811277}	[0.47596879781640905, 0.4759688005738675, 0.47...	[0.006397914886474609, 0.007554340362548828, 0...	2223006.586	1.296480e+13	3600666.356	0.691	1273741.727
ElasticNet	0.563065	{'elasticnet_alpha': 0.2782559402207126, 'ela...	[0.3570412947232106, 0.3570412947232106, 0.357...	[0.0068953514099121095, 0.0059603691101074215,...	2440616.382	1.419283e+13	3767336.608	0.662	1603599.745
DecisionTree	0.484125	{'decisiontreeregressor_max_depth': 4, 'decis...	[0.1337091902292908, 0.1337091902292908, 0.133...	[0.0069405555725097655, 0.007053375244140625, ...	2265857.068	1.274568e+13	3570109.299	0.697	1335396.713
RandomForestRegressor	0.607108	{'randomforestreregessor_bootstrap': True, 'ra...	[0.1444604448103355, 0.13764872122730665, 0.14...	[0.014247560501098632, 0.023487043380737305, 0...	2155296.169	1.189030e+13	3448232.004	0.717	1283422.783
XGBRegressor	0.653669	{'xgbregressor_learning_rate': 0.08, 'xgbregr...	[0.6208600415906375, 0.6414777103665003, 0.650...	[0.011018323898315429, 0.01084585189819336, 0....	1898797.729	9.970768e+12	3157652.348	0.763	1015940.000

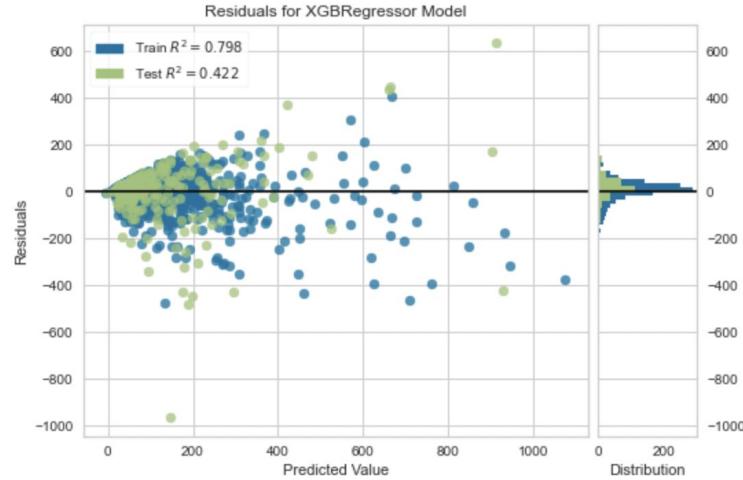
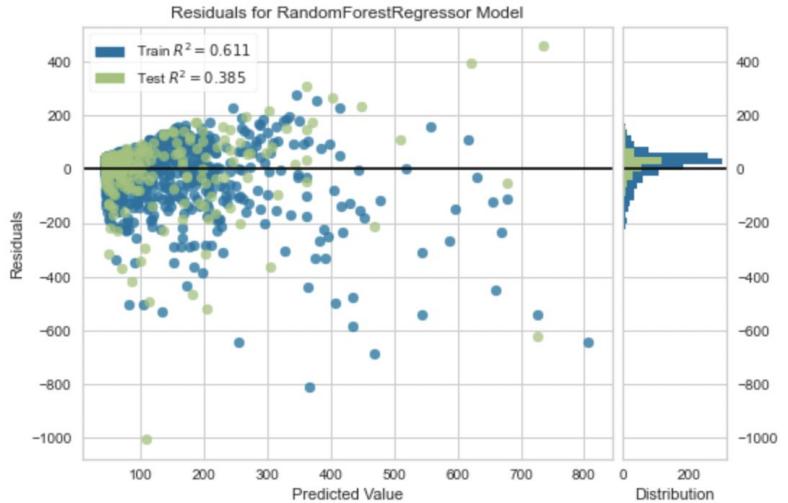
# Comparer les modèles \_ SiteEnergyUse



Le choix se portera sur des modèles ensemblistes et plus particulièrement, l'algorithme de RandomForest et XGBoost

# Evaluation du modèle \_ TotalGHGEmissions

Les distributions des résidus du jeu d'entraînement et du jeu de test semblent ne suivre pas une loi normale. Puisque le nombre de données est moins important (seulement 20% de l'échantillon) on voit que la distribution des résidus du jeu de test est plus aplatie.

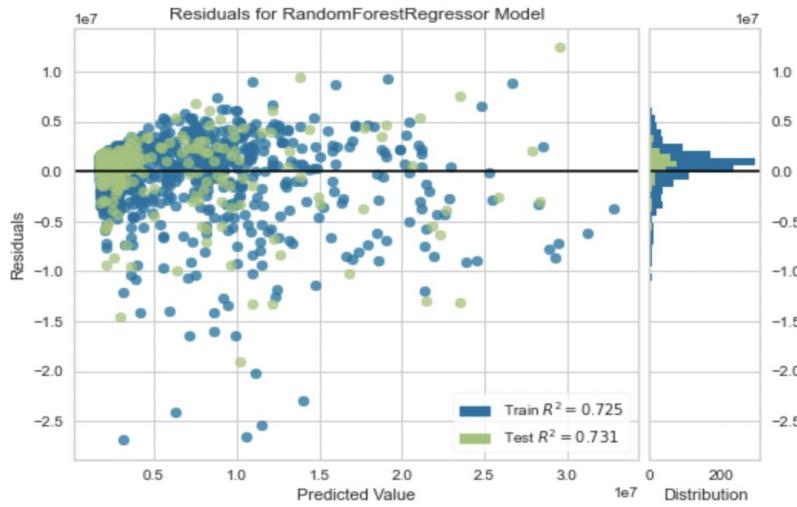


```
final model RandomForest
MAE: 70.40393
RMSE: 120.95297846684058
R2: 0.38486
-----
final model XGBOOST
MAE: 65.21701
RMSE: 120.95297846684058
R2: 0.42204
```

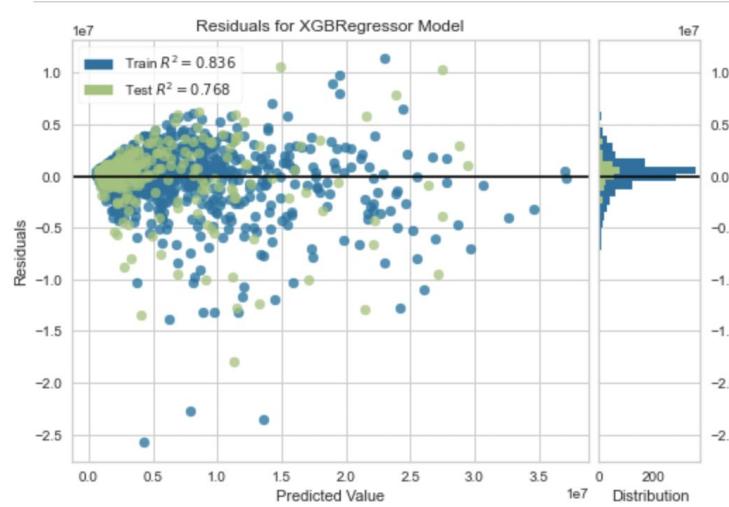
La différence entre le coefficient de détermination du jeu d'entraînement et du jeu de test montre qu'il y a overfitting de notre modèle malgré les précautions prises validation croisée, contrainte du modèle en utilisant les paramètres.

Modèle XGBOOST permet d'expliquer 80% de la variance expliquée mais les autres indicateurs de performances sont moins bons que pour la prédiction de la consommation d'énergie

# Evaluation du modèle \_ SiteEnergyUse



```
final model RandomForest
MAE: 2084363.81633
RMSE: 2774728.149231806
R2: 0.69624
-----
final model XGBOOST
MAE: 1572353.20982
RMSE: 2774728.149231806
R2: 0.79255
```

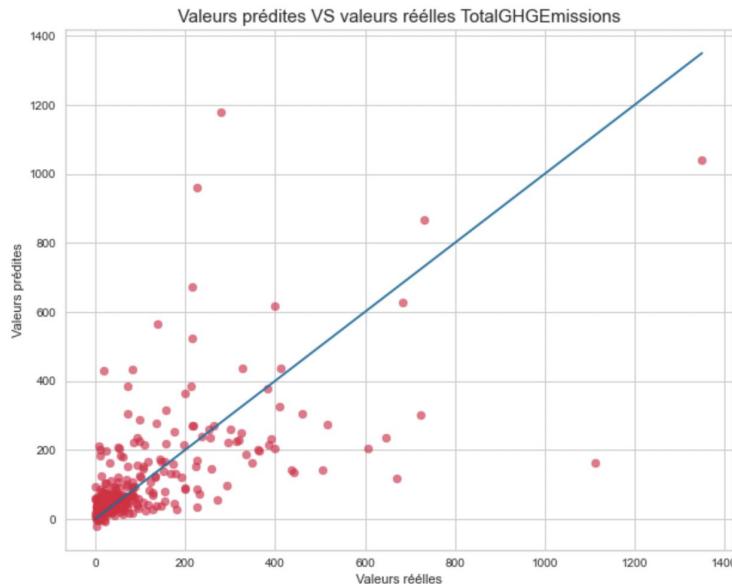


Modèle XGBOOST permet d'expliquer 84% de la variance expliquée

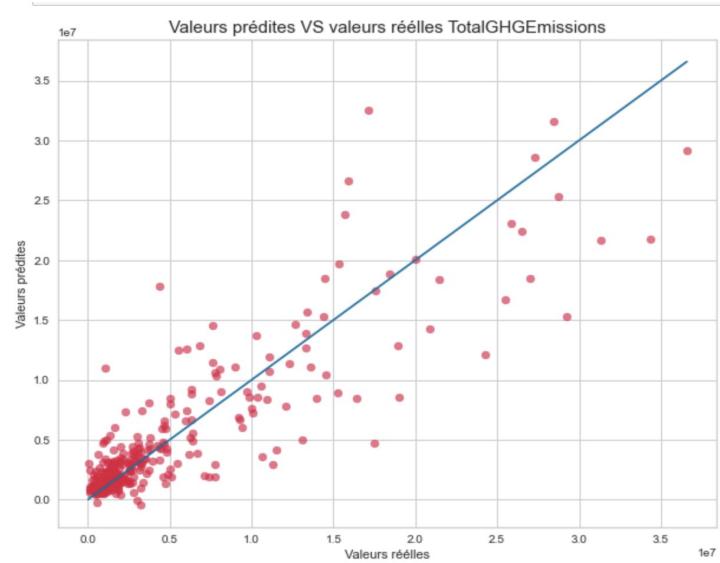
Modèle Random Forest permet d'expliquer 73% de la variance expliquée

# Des valeurs prédictes vers valeurs réelles

## TotalGHGEmissions



## SiteEnergyUse



Les valeurs prédictes sont un peu éloignées de la première bissectrice. En effet, les écarts et mauvais résultats obtenus dépendent donc du faible nombre de données qui impactent le Train\_Test\_Split initial. Le modèle est correctement entraîné mais n'obtient pas de très bon résultats sur le jeu de test (pas d'overfitting constaté dans les entraînements).

# Apport de l'energy star score

- Supprimer les lignes où l'energy star score est manquant
- Entraîner le meilleur modèle sans la variable energy star score
- Entraîner le meilleur modèle avec la variable energy star score

TotalGHGEmissions

model	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
Dummy Regressor	95.553	37933.265	194.765	-0.160	36.048
SVR	88.979	35104.962	187.363	-0.074	30.085
DecisionTree	89.077	26820.020	163.768	0.180	43.250
ElasticNet	80.442	21493.025	146.605	0.343	46.755
Lasso	71.597	16972.038	130.277	0.481	36.262
Ridge Kernel	68.122	15030.438	122.599	0.540	35.985
Ridge	67.910	14902.525	122.076	0.544	36.890
RandomForestRegressor	62.529	13416.664	115.830	0.590	30.257
GradientBoosting	58.097	11825.111	108.743	0.638	27.201
XGBRegressor	60.298	11246.768	106.051	0.656	28.041

SiteEnergyUse

model	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
Dummy Regressor	95.553	37933.265	194.765	-0.160	36.048
SVR	88.979	35104.962	187.363	-0.074	30.085
DecisionTree	89.077	26820.020	163.768	0.180	43.250
ElasticNet	80.442	21493.025	146.605	0.343	46.755
Lasso	71.597	16972.038	130.277	0.481	36.262
Ridge Kernel	68.122	15030.438	122.599	0.540	35.985
Ridge	67.910	14902.525	122.076	0.544	36.890
RandomForestRegressor	62.529	13416.664	115.830	0.590	30.257
GradientBoosting	58.097	11825.111	108.743	0.638	27.201
XGBRegressor	60.298	11246.768	106.051	0.656	28.041

# Apport de l'energy star score en fixant paramètres\_TotalGHGEmissions

## Avec l'energy Star Score

model	best_score	best_params	mean_test_score	mean_score_time	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
XGBRegressor	0.470825	{'xgbregressor__learning_rate': 0.05, 'xgbregressor__max_depth': 4, 'xgbregressor__n_estimators': 100}	[0.4391119367574827, 0.43904914056804156, 0.39...]	[0.006965017318725586, 0.01156129837036132, 0...]	60.668	11996.347	109.528	0.633	29.208
RandomForestRegressor	0.452428	{'randomforestregressor__bootstrap': True, 'randomforestregressor__max_depth': 3, 'randomforestregressor__n_estimators': 100}	[0.1092508023990447, 0.10845278107665082, 0.10...]	[0.009708023071289063, 0.01236419677734375, 0....]	67.317	14151.126	118.959	0.567	34.765
Ridge	0.410932	{'ridge__alpha': 46.41588833612782, 'ridge__copy_X': False, 'ridge__fit_intercept': True, 'ridge__solver': 'cholesky'}	[0.3363041864507589, 0.33613562893016075, 0.33...]	[0.004350423812866211, 0.003808403015136719, 0...]	69.949	16377.318	127.974	0.499	39.909
Lasso	0.360522	{'lasso__alpha': 0.2782559402207126}	[0.30922606261457214, 0.3092795275044755, 0.30...]	[0.0057752132415771484, 0.005241537094116211, ...]	70.362	16529.947	128.569	0.494	39.222
ElasticNet	0.359637	{'elasticnet__alpha': 0.2782559402207126, 'elasticnet__l1_ratio': 0.5}	[0.2500634186578824, 0.2500634186578824, 0.250...]	[0.008170175552368163, 0.004286813735961914, 0...]	76.027	19237.996	138.701	0.412	44.970
DecisionTree	0.247981	{'decisiontreeregressor__max_depth': 3, 'decisiontreeregressor__min_samples_leaf': 1, 'decisiontreeregressor__min_weight_fraction_leaf': 0.001, 'decisiontreeregressor__splitter': 'best'}	[0.05276317836599444, 0.05276317836599444, 0...]	[0.0062195777893066405, 0.003697109222412109, ...]	76.513	18514.432	136.068	0.434	33.429

## Sans energy Star Score

model	best_score	best_params	mean_test_score	mean_score_time	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
XGBRegressor	0.459060	{'xgbregressor__learning_rate': 0.05, 'xgbregressor__max_depth': 4, 'xgbregressor__n_estimators': 100}	[0.40434277792838935, 0.42741611449099537, 0.4...]	[0.010956239700317384, 0.010713958740234375, 0...]	65.217	14629.623	120.953	0.422	29.443
Ridge	0.398844	{'ridge__alpha': 3.593813663804626, 'ridge__copy_X': False, 'ridge__fit_intercept': True, 'ridge__solver': 'cholesky'}	[0.24646927745244174, 0.24716007658165395, 0.2...]	[0.006150674819946289, 0.005318260192871094, 0...]	73.435	16621.990	128.926	0.343	35.466
RandomForestRegressor	0.360241	{'randomforestregressor__bootstrap': True, 'randomforestregressor__max_depth': 3, 'randomforestregressor__n_estimators': 100}	[0.06490053136937587, 0.06622847895968616, 0.0...]	[0.008896827697753906, 0.01146559715270996, 0....]	70.237	15671.695	125.187	0.381	39.021
Lasso	0.358209	{'lasso__alpha': 0.2782559402207126}	[0.24156085089199805, 0.24163154425765718, 0.2...]	[0.005522727966308594, 0.0047207832336425785, ...]	74.452	16535.880	128.592	0.347	39.540
ElasticNet	0.320966	{'elasticnet__alpha': 0.021544346900318846, 'elasticnet__l1_ratio': 0.5}	[0.17941622476346225, 0.17941622476346225, 0.1...]	[0.007697343826293945, 0.005574703216552734, 0...]	75.240	16376.136	127.969	0.353	38.543
DecisionTree	0.186262	{'decisiontreeregressor__max_depth': 3, 'decisiontreeregressor__min_samples_leaf': 1, 'decisiontreeregressor__min_weight_fraction_leaf': 0.001, 'decisiontreeregressor__splitter': 'best'}	[0.03497041783250412, 0.03497041783250412, 0.0...]	[0.0058135509490966795, 0.0041691303253173825,...]	77.158	19166.747	138.444	0.243	46.851

# Apport de l'energy star score en fixant paramètres \_ SiteEnergyUse

## Avec l'energy Star Score

model	best_score	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
XGBRegressor	0.807052	1659881.867	7.699116e+12	2774728.149	0.829	844579.781
RandomForestRegressor	0.747287	2054568.838	1.071013e+13	3272634.210	0.762	1122091.520
Ridge	0.745296	1780328.842	8.755140e+12	2958908.642	0.806	1053926.537
Lasso	0.684299	2167623.538	1.266136e+13	3558279.964	0.719	1234265.376
ElasticNet	0.674998	2403790.810	1.440922e+13	3795948.380	0.680	1353631.134
DecisionTree	0.630539	2436344.641	1.525254e+13	3905450.062	0.662	1173002.935

## Sans energy Star Score

model	best_score	MAE	MSE	RMSE	R <sup>2</sup>	median abs err
XGBRegressor	0.653669	1898797.729	9.970768e+12	3157652.348	0.763	1015940.000
Ridge	0.649723	2185841.789	1.276844e+13	3573295.912	0.696	1194681.702
Lasso	0.612418	2223006.586	1.296480e+13	3600666.356	0.691	1273741.727
RandomForestRegressor	0.607108	2155296.169	1.189030e+13	3448232.004	0.717	1283422.783
ElasticNet	0.563065	2440616.382	1.419283e+13	3767336.608	0.662	1603599.745
DecisionTree	0.484125	2265857.068	1.274568e+13	3570109.299	0.697	1335396.713

# Apport de l'energy star score pour 2 modèles avec best paramètres

## TotalGHGEmissions

	model	best score	column Energy star	mean_test_score	mean_score_time	mean_fit_time	RMSE	MSE	MAE	R <sup>2</sup>	median abs err
0	RandomForestRegressor	0.364	False	[0.364]	[0.011]	[0.471]	70.404	15570.621	124.782	0.385	39.099
1	XGBRegressor	0.459	False	[0.459]	[0.011]	[2.671]	65.217	14629.623	120.953	0.422	29.443
2	RandomForestRegressor	0.451	True	[0.451]	[0.016]	[0.456]	67.745	14384.691	119.936	0.560	34.456
3	XGBRegressor	0.460	True	[0.46]	[0.01]	[1.538]	59.131	12111.764	110.053	0.630	28.570

## SiteEnergyUse

	model	best score	column Energy star	mean_test_score	mean_score_time	mean_fit_time	RMSE	MSE	MAE	R <sup>2</sup>	median abs err
0	RandomForestRegressor	0.589	False	[0.589]	[0.012]	[0.475]	2086024.764	1.128893e+13	3359899.727	0.731	1200388.301
1	XGBRegressor	0.646	False	[0.646]	[0.019]	[2.735]	1848758.744	9.762449e+12	3124491.833	0.768	938759.812
2	RandomForestRegressor	0.733	True	[0.733]	[0.011]	[0.369]	2027130.405	1.072061e+13	3274233.733	0.762	1138645.418
3	XGBRegressor	0.792	True	[0.792]	[0.007]	[0.921]	1705698.938	8.794062e+12	2965478.449	0.805	758526.750

La variable cible étant la même et les variables explicatives presque identiques, on se retrouve avec des résultats qui sont proches de ce qui avait été vu auparavant.

# Conclusion

- ❖ Les informations dont l'on dispose nous permettent d'avoir une prédiction de la consommation d'énergie
- ❖ Le XGBRegresson et le RandomForestRegressor sont nos deux algorithmes les plus performants
- ❖ On a cherché à optimiser les paramètres de ces différents algorithmes par le biais d'une validation croisée
- ❖ On remarque que le type d'usage est important dans nos prédictions
- ❖ La différence entre le coefficient de détermination du jeu d'entraînement et du jeu de test montre qu'il y a overfitting de notre modèle malgré les précautions prises validation croisée, contrainte du modèle en utilisant les paramètres. Le modèle XGBOOST permet d'expliquer 83% de la variance expliquée.
- ❖ Ces modèles peuvent être utilisés pour nos prédictions mais il faudra prendre en compte que les erreurs sont plus importantes que pour la consommation d'énergie d'un bâtiment.
- ❖ L'ajout d'une variable supplémentaire comme le score Energy Star ne va pas modifier les scores de la prédiction.

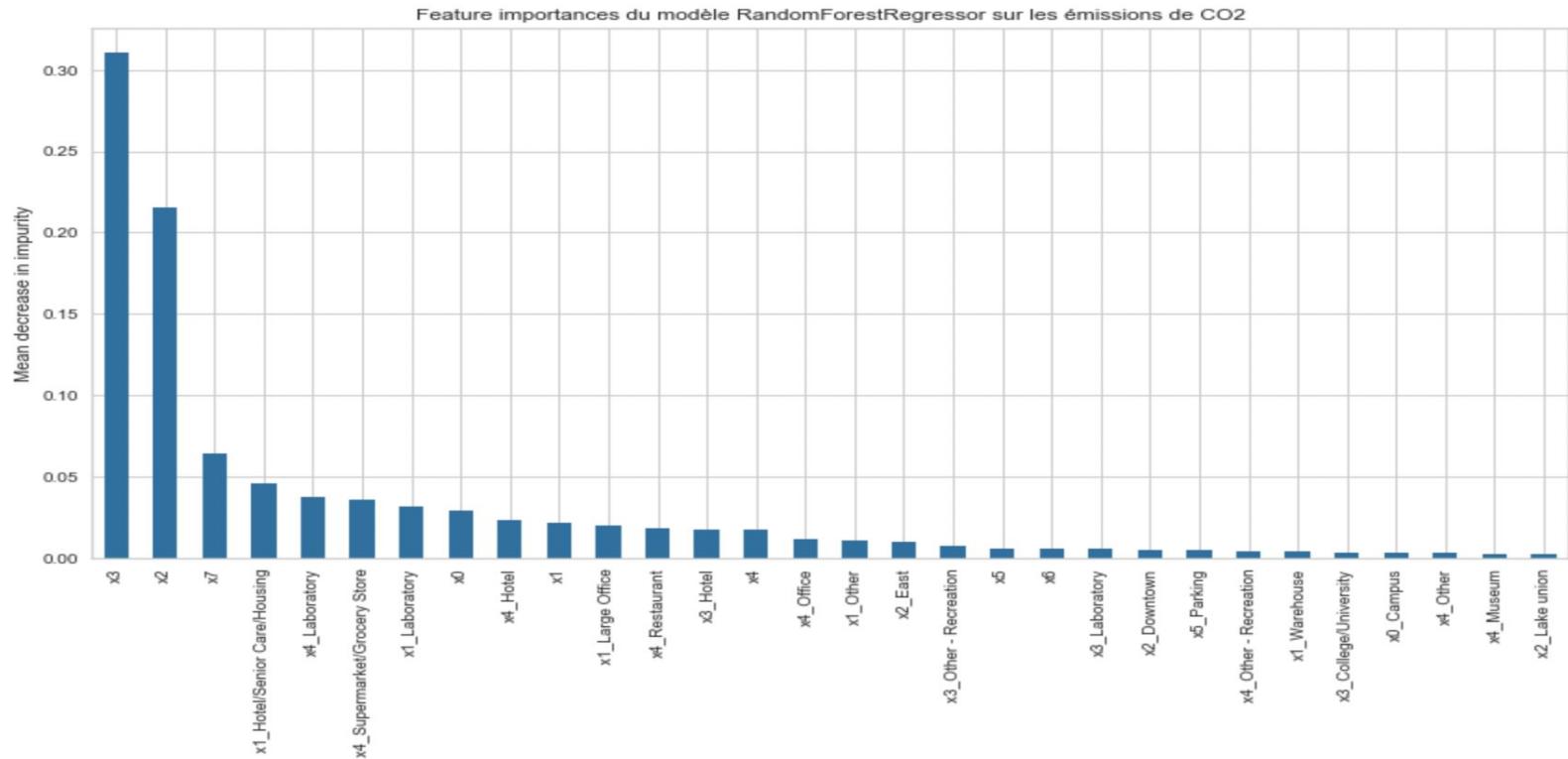


# Améliorations

- ❖ Création d'une API web pour mettre à disposition le meilleur modèle
- ❖ Déployer le modèle sur un service cloud
- ❖ Recueillir des données plus récentes
- ❖ Combiner avec des historiques météo



# Bonus: Importance des variables \_ TotalGHGEmissions



# Importance des variables \_ SiteEnergyUse

