Projet 3

Anticipez la consommation électrique de bâtiments

Objectif

La ville de Seattle souhaite atteindre la neutralité carbone d ici 2050

Pour cela des relevés par des agents ont été exécutés en 2015 et 2016. Ces relevés sont fastidieux et demandent beaucoup de travail, la ville voudrait tenter de prédire les données pour les bâtiments non destinés à l'habitation et non encore relevés.

Objectif : Prédire pour les bâtiments non destinés à l'habitation :

- les besoins en consommation énergétique
- les émissions de CO2

Évaluer l'intérêt de l'Energy Star Score

Le score « ENERGY STAR » étant calculé en fonction de la consommation d'énergie, de l'utilisation du bâtiment, on pourrait envisager d'utiliser la prédiction de consommation d'énergie pour prédire l'émission de CO2. Nous ferons une comparaison avec et sans cette donnée.

Variable à prédire

Deux variables cibles :

- consommation d’energie,
- émission de CO2

Nous avons les jeux de données 2015 et 2016 contenant un peu plus de 3000 lignes et 46 colonnes. Les bâtiments sont identifiés par un numéro unique ce qui va permettre de fusionner les deux fichiers.

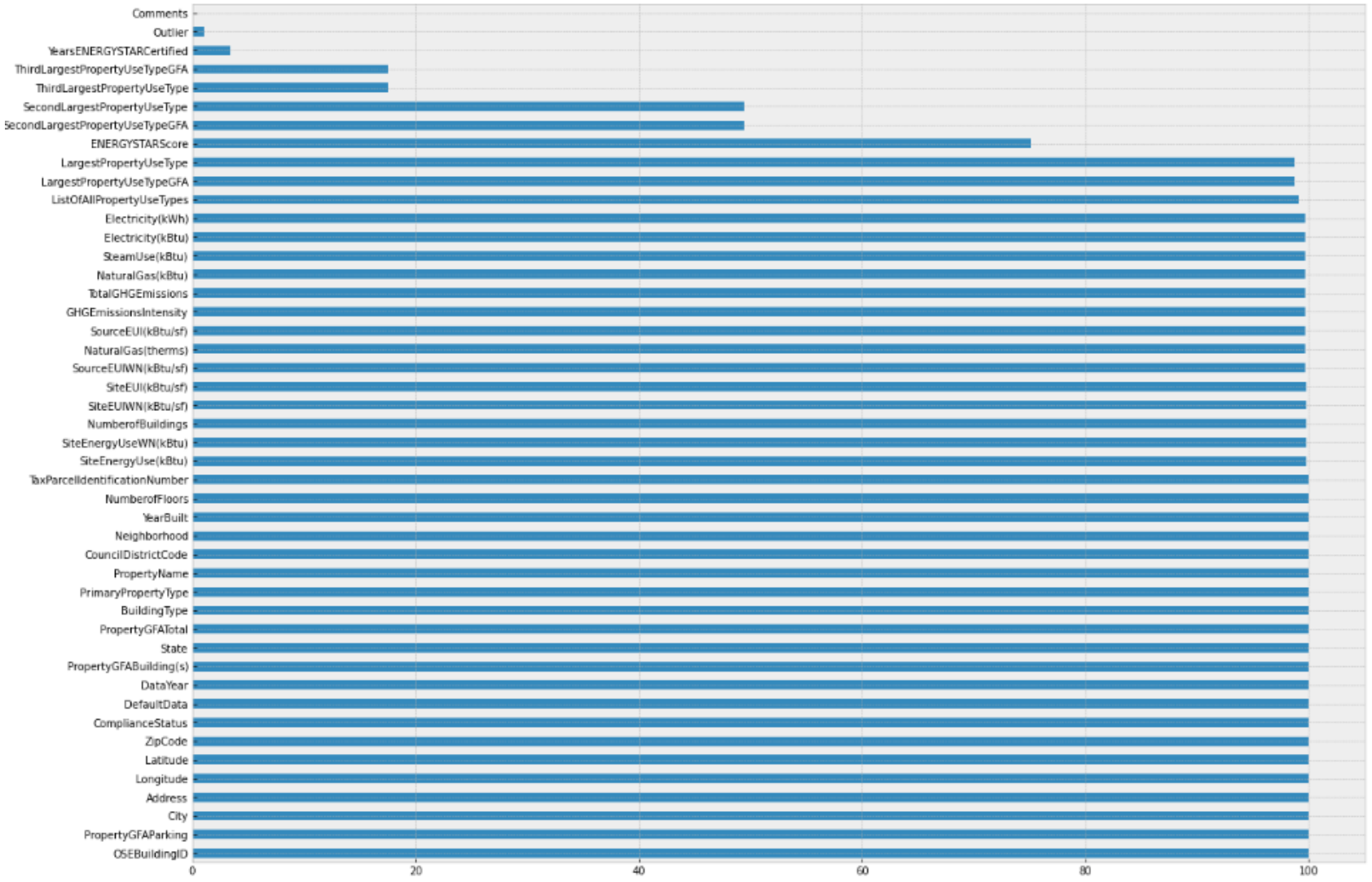
Nous allons garder les valeurs les plus hautes pour un bâtiment lors de la fusion

Peu de valeur manquantes pour les features que nous allons utiliser sauf EnergyStarScore (remplie dans 72% des données)

Assemblage cleanage des données

- → harmonisation des données 2015 et 2016
- → suppression/renommage des colonnes non communes
- → récupération de plusieurs variables splitter
- → dédoublonnage par bâtiments en gardant les valeurs les plus grandes
- → features engineering sur les variables catégorielle
- → fusion des dataframes
- → Restriction du jeu de données sur les bâtiments non destinés à l habitation

Taux de remplissage du jeu de donnée



Étape global

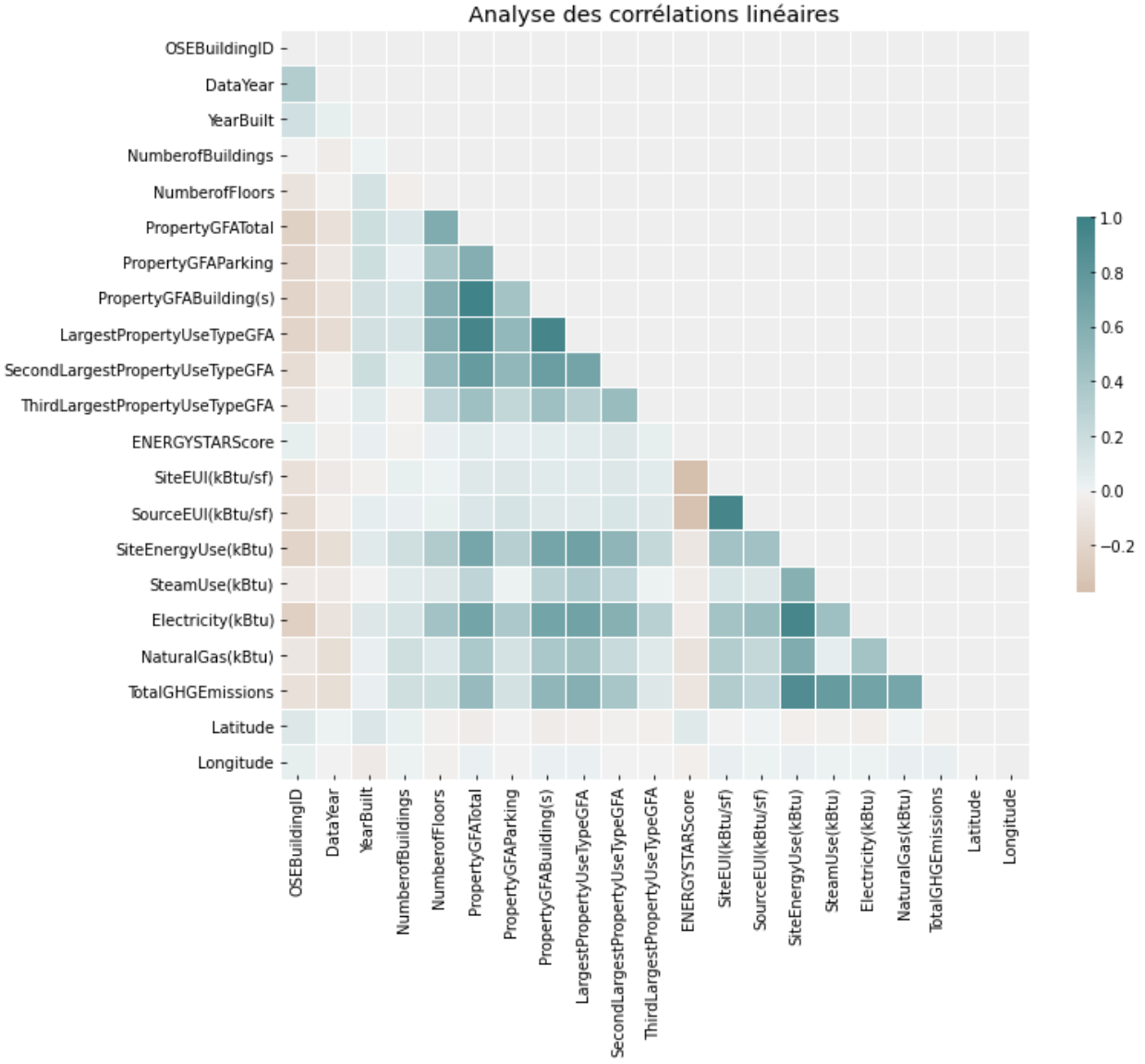
Nettoyage/dédoublonnage

Analyse

Modélisation

Évaluation des performances

Corrélation de nos variables



Corrélation de nos variables

la régression linéaire souffre de quelques inconvénients quand les variables sont corrélées : la solution n'est pas unique et les coefficients ont une grande variabilité, et l'interprétation est plus difficile

Pour éviter cela nous allons calculer de nouvelles propriétés grâce à nos informations déjà existantes , comme l age du bâtiment , ou la surface par étage..
Et ensuite nous supprimerons nos variables trop corrélées.

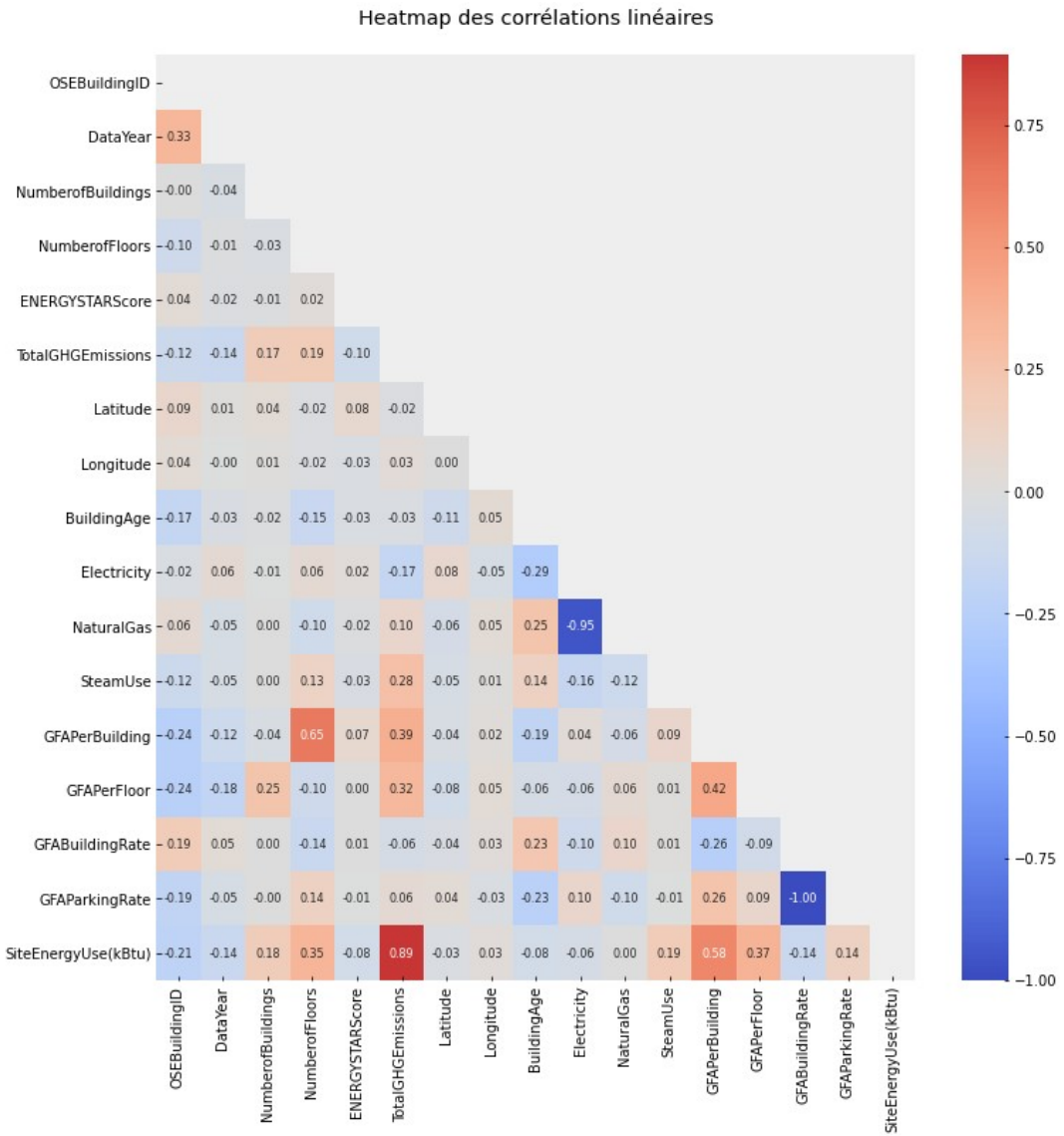
Nous avons besoins de quelques hypothèses pour nos valeurs à traiter :

la première, l'hypothèse de **linéarité**

la deuxième, l'hypothèse de **normalité**

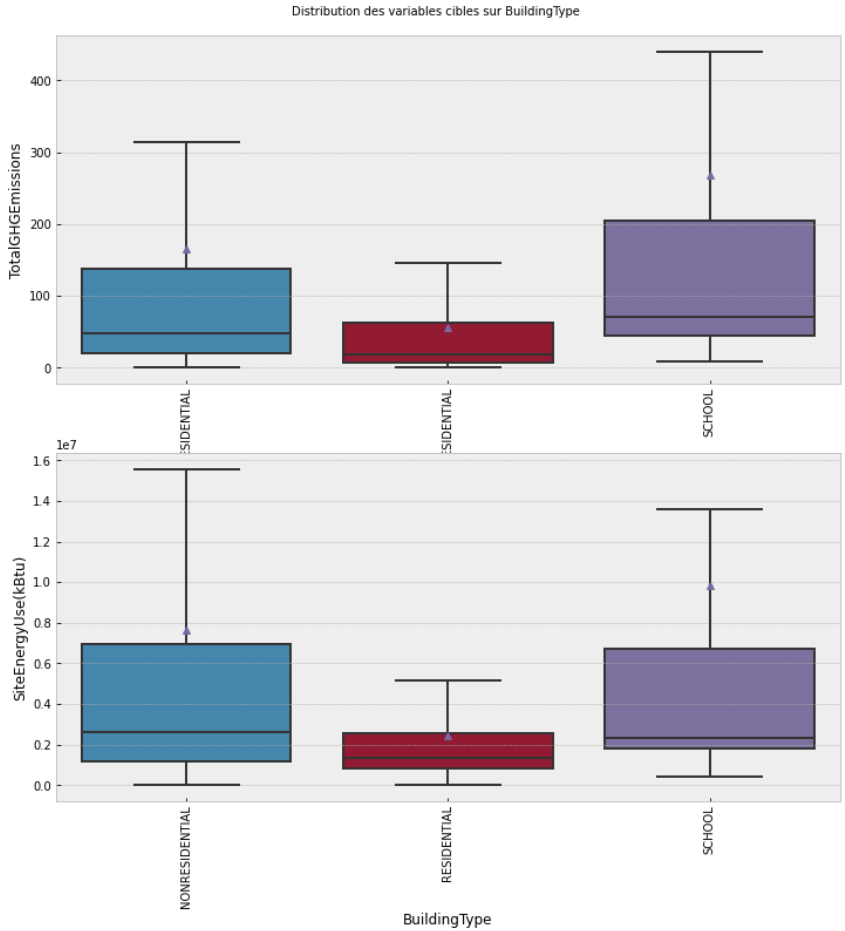
la troisième, l'hypothèse **d'indépendance**

Corrélation après traitement



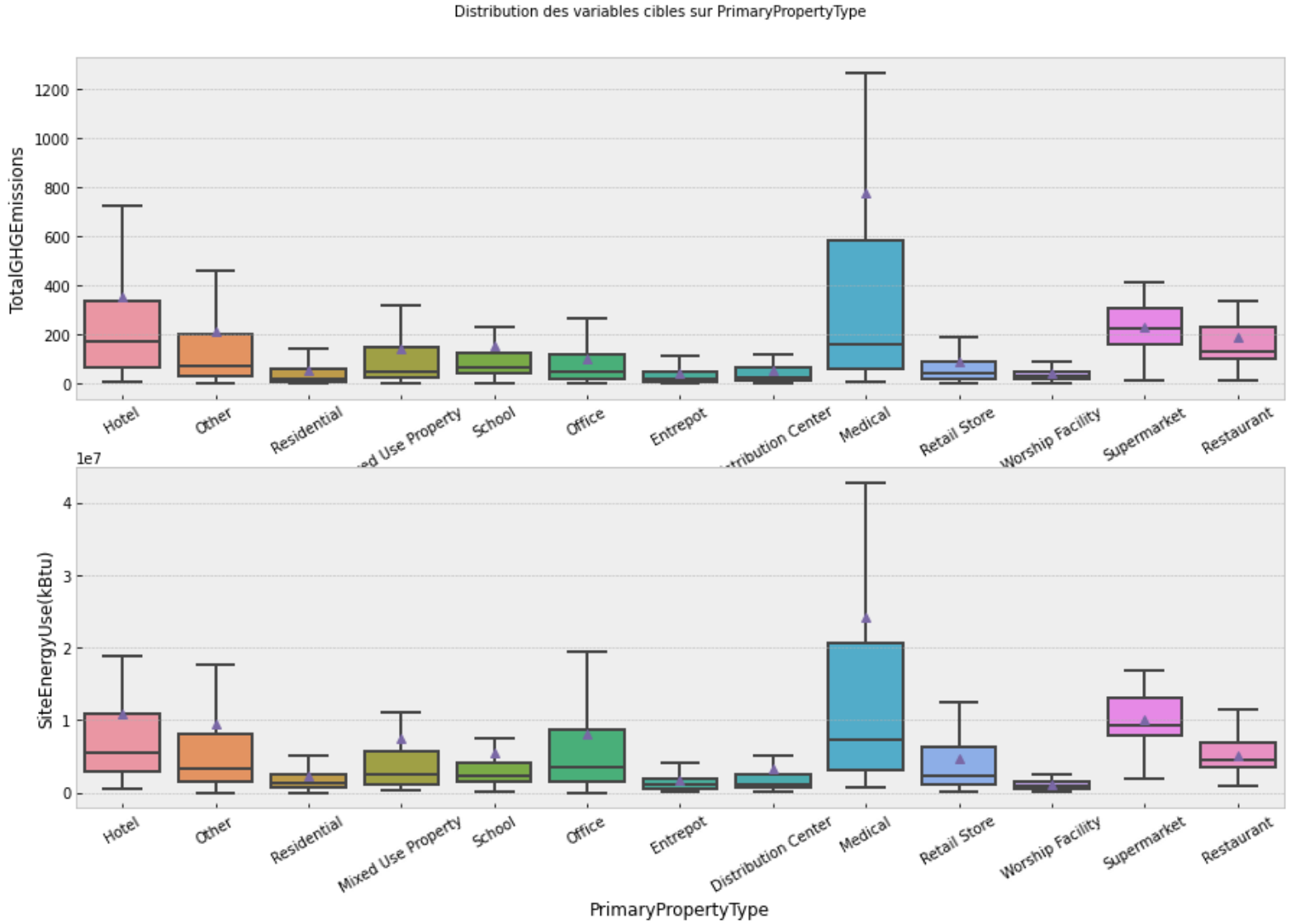
Analyse multivariées

Le secteur de l éducation sont de grands consommateurs d électricité et émetteur de co2



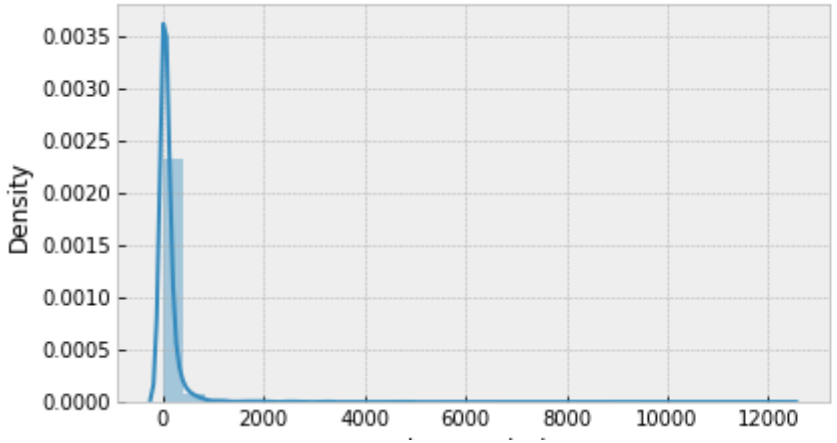
Analyse multivariées

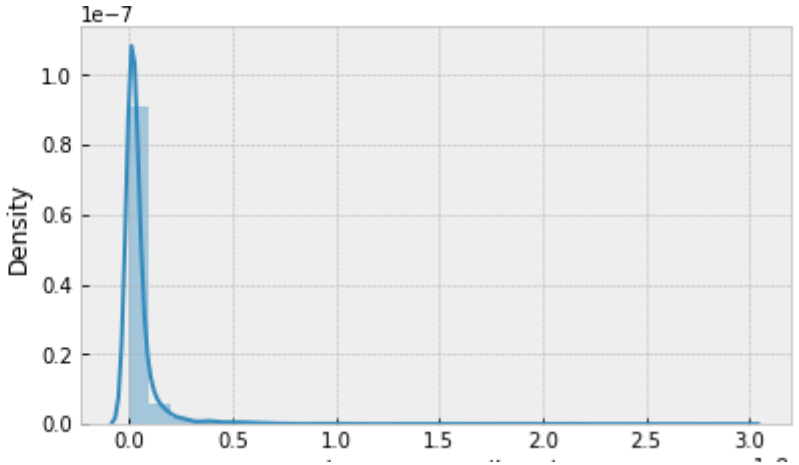
Le secteur de la santé sont de grands consommateurs d électricité et émetteur de co2

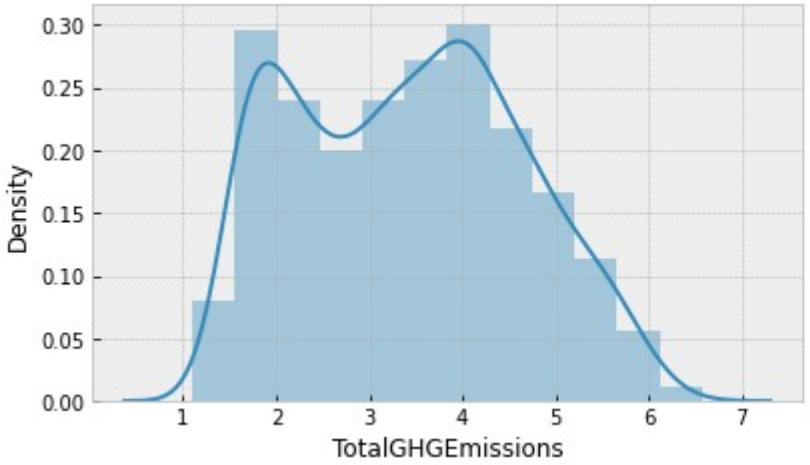


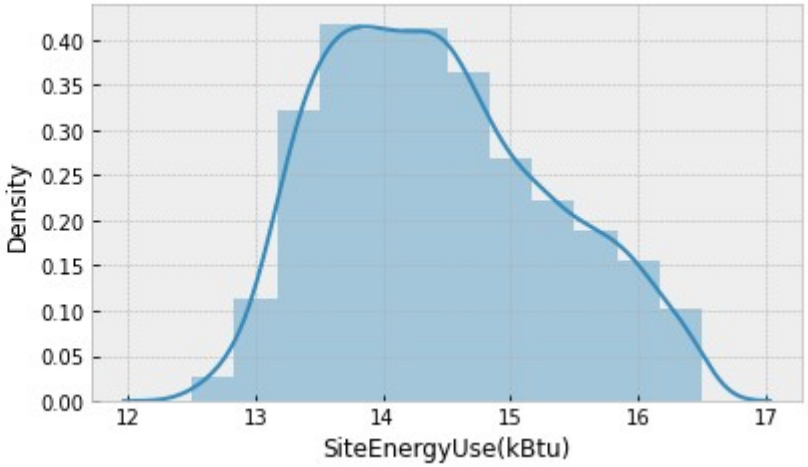
Passage au log

Aucune variable n'a de distribution normale









Data processing

TargetEncoder : Le target encoding consiste à remplacer la valeur de chaque variable catégorielle par la moyenne de la cible des individus ayant la même valeur pour la variable catégorielle.

Robust Scaler : Normalise nos données et gère les outliers mieux que StandarScaler

Indicateur boolean : sur les autres types d énergie (gaz, steamuse)

passage au log nos features avant l entraînement

Création d’un Train set et Test set pour comparer les modèles

Étape de modalisation

Data processing
Transformation de nos variables et variable cibles

Modèle par défaut
Exécution des modèles par défaut

Cross validation
Mise en place d une validation croisée kfold 5

Hyperparametres GridSearch
Optimisation de nos paramètres

Modèles choisis

- LinearRegression,
- DummyRegressor,

Base line

- ElasticNet,
- Ridge,
- Lasso,
- SVR,

Régularisation

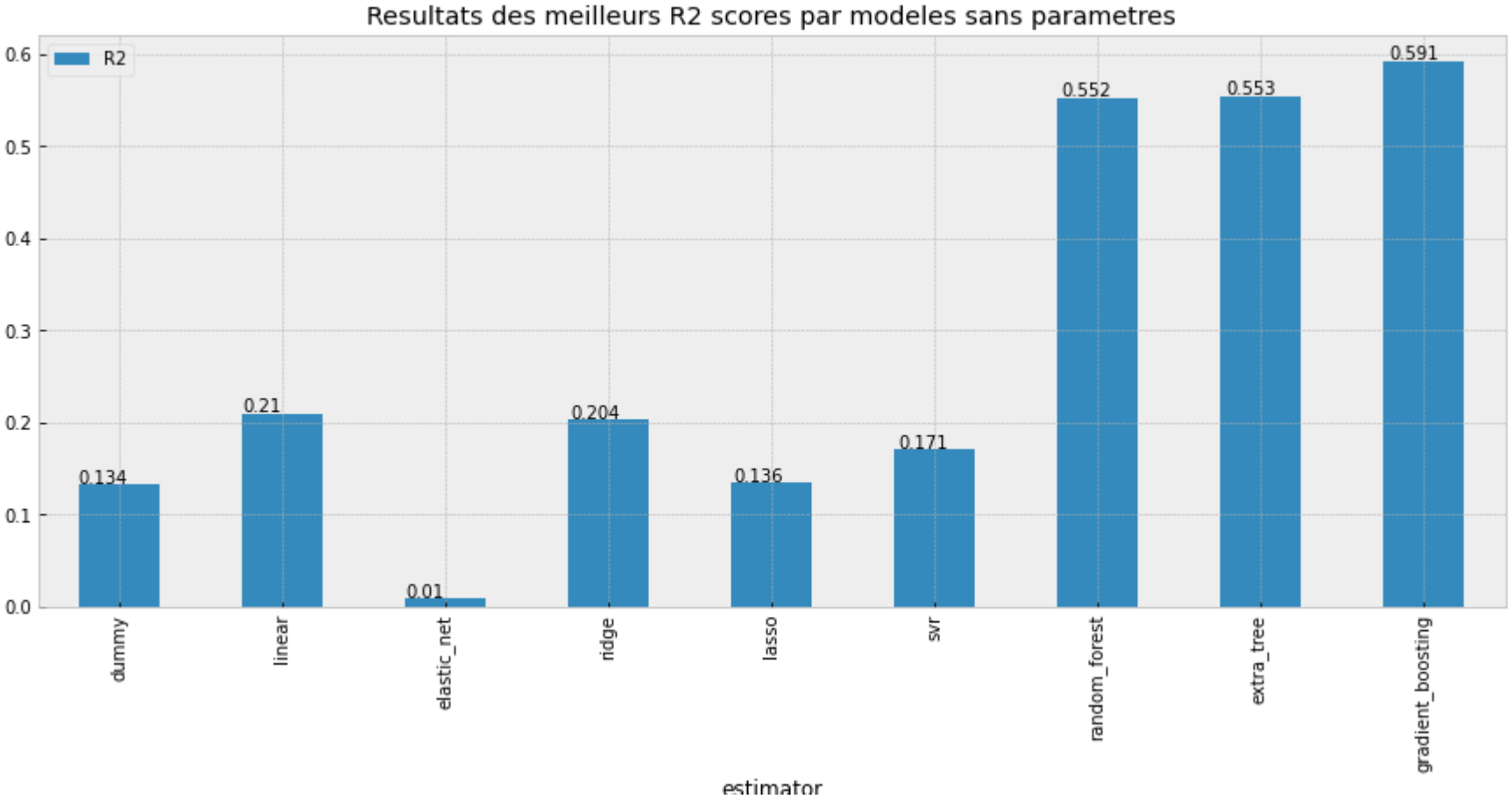
- RandomForestRegressor,
- ExtraTreesRegressor,
- GradientBoostingRegressor

Méthode ensembliste

Métriques sélectionnées

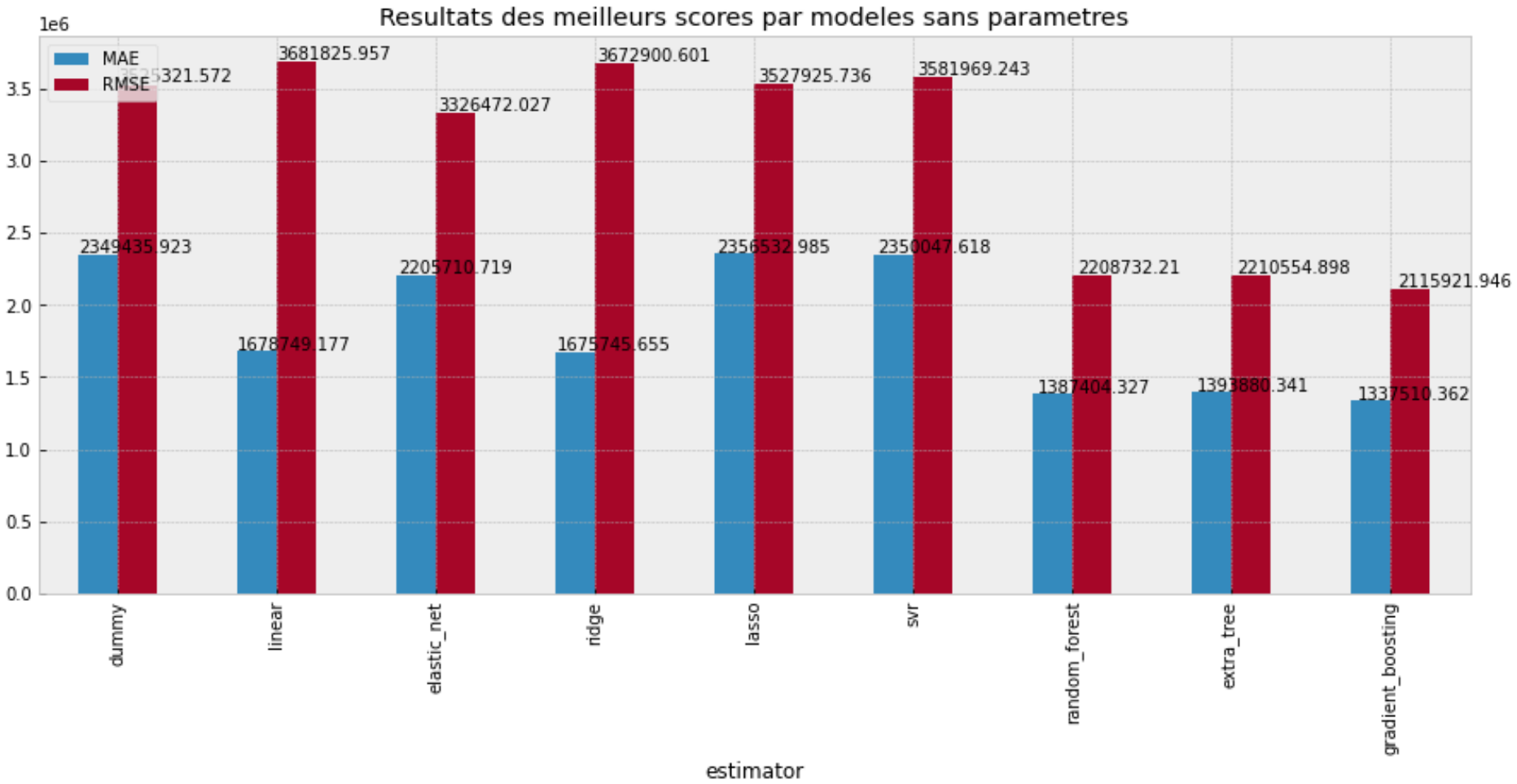
- R2
- MAE
- RMSE

Résultat en mode par défaut



Résultat en mode par défaut

Résultat des scores pour l énergie



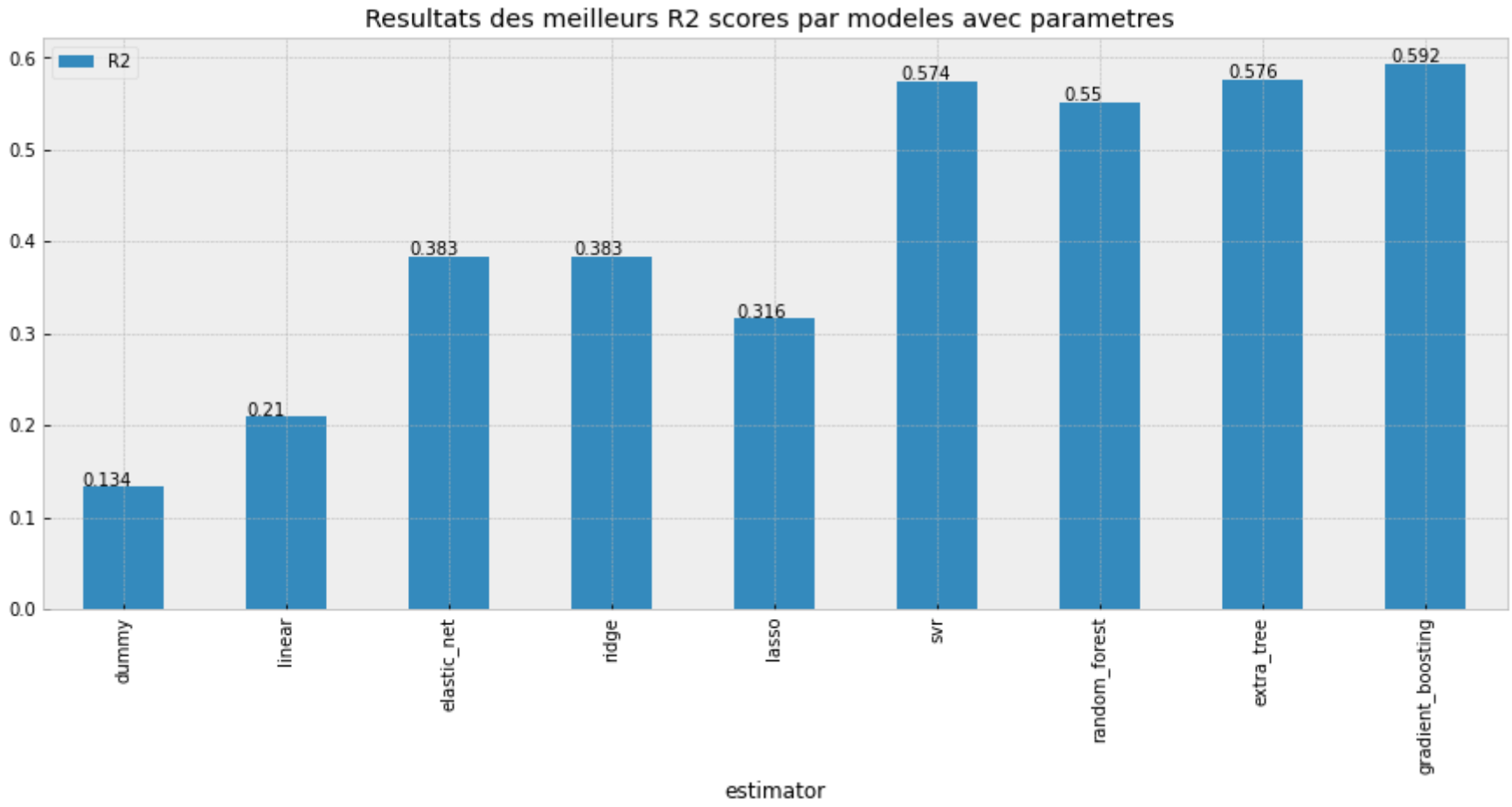
Trois modèles ont les meilleurs scores après cross validation random Forest , extra tree et gradient boosting

Meilleurs paramètres gridSearch

Investigation afin de trouver les meilleurs paramètres
Un randomSearch pour récupérer des ranges efficaces, puis un gridSearch afin de détecter les meilleurs paramètres de nos ranges

estimator	best_params
dummy	{'dummy__regressor__strategy': 'mean'}
linear	{'linear__regressor__copy_X': True, 'linear__regressor__fit_intercept': True}
elastic_net	{'elastic_net__regressor__alpha': 0.9000000000000001, 'elastic_net__regressor__l1_ratio': 0.0, 'elastic_net__regressor__max_iter': 2000, 'elastic_net__regressor__selection': 'random'}
ridge	{'ridge__regressor__alpha': 756.463327554629, 'ridge__regressor__max_iter': 1000}
lasso	{'lasso__regressor__alpha': 0.1747528400007683, 'lasso__regressor__max_iter': 1000}
svr	{'svr__regressor__C': 0.5, 'svr__regressor__epsilon': 0.1, 'svr__regressor__gamma': 'auto', 'svr__regressor__kernel': 'rbf'}
random_forest	{'random_forest__regressor__max_features': 'sqrt', 'random_forest__regressor__n_estimators': 80}
extra_tree	{'extra_tree__regressor__bootstrap': True, 'extra_tree__regressor__min_samples_split': 4, 'extra_tree__regressor__n_estimators': 80}
gradient_boosting	{'gradient_boosting__regressor__learning_rate': 0.01, 'gradient_boosting__regressor__max_leaf_nodes': 20, 'gradient_boosting__regressor__n_estimators': 1000}

Résultat après optimisation

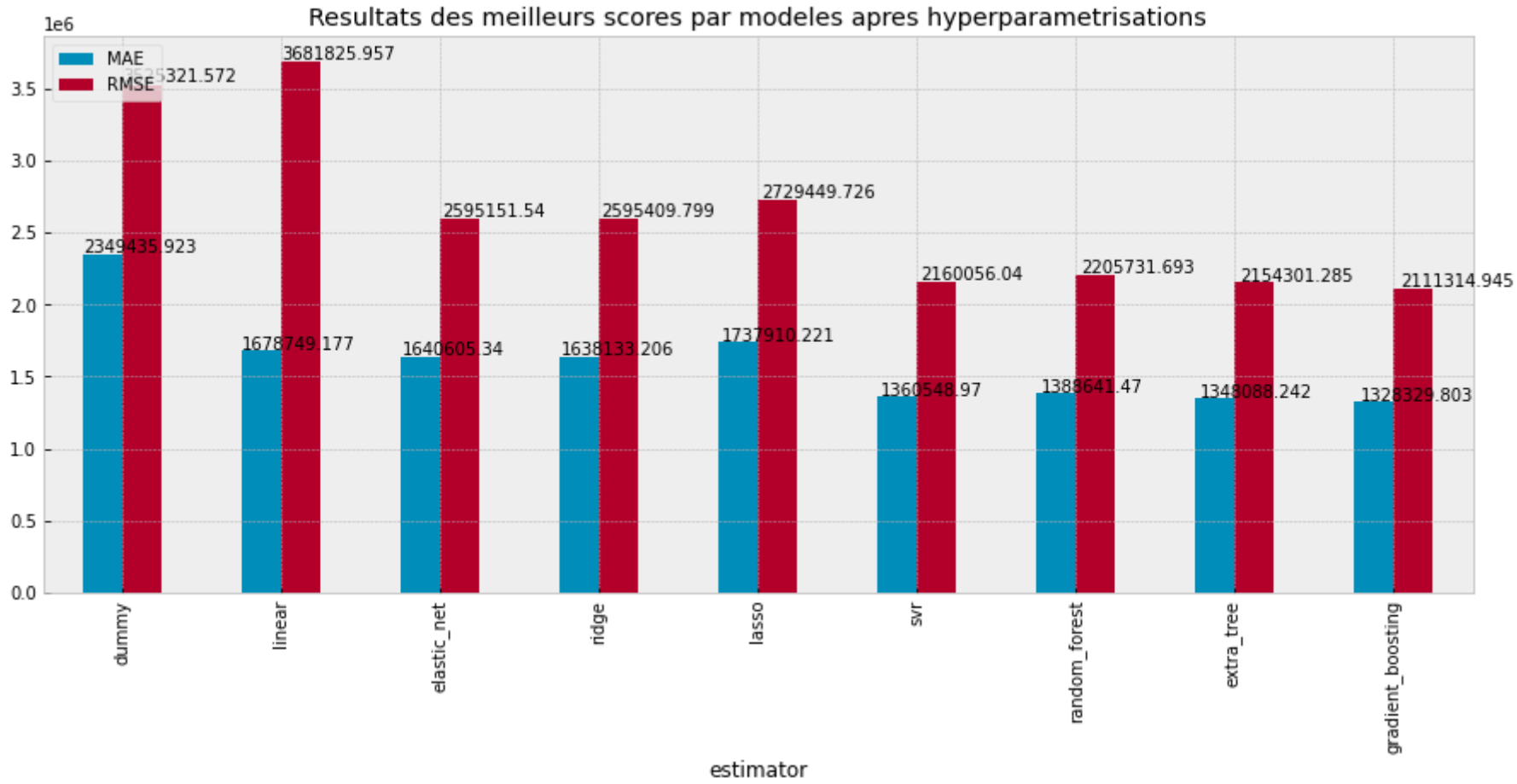


Résultat après optimisation

estimator		R2	MAE	RMSE
elastic_net	with options	0.383	1640605.340	2595151.540
	default	0.010	2205710.719	3326472.027
ridge	with options	0.383	1638133.206	2595409.799
	default	0.204	1675745.655	3672900.601
lasso	with options	0.316	1737910.221	2729449.726
	default	0.136	2356532.985	3527925.736
svr	with options	0.574	1360548.970	2160056.040
	default	0.171	2350047.618	3581969.243
random_forest	with options	0.550	1394085.970	2215150.441
	default	0.552	1387404.327	2208732.210
extra_tree	with options	0.576	1339543.039	2155371.329
	default	0.553	1393880.341	2210554.898
gradient_boosting	with options	0.592	1328244.673	2111652.255
	default	0.591	1337510.362	2115921.946

Résultat sur les prédictions

Nos trois méthodes ensemblistes ont les meilleurs résultats



Détails des résultats par types de modèles

Underfitting

• Erreur d'entrainement élevé
• Erreur d'entrainement proche de l'erreur de test
• Biais élevé

Remède possible

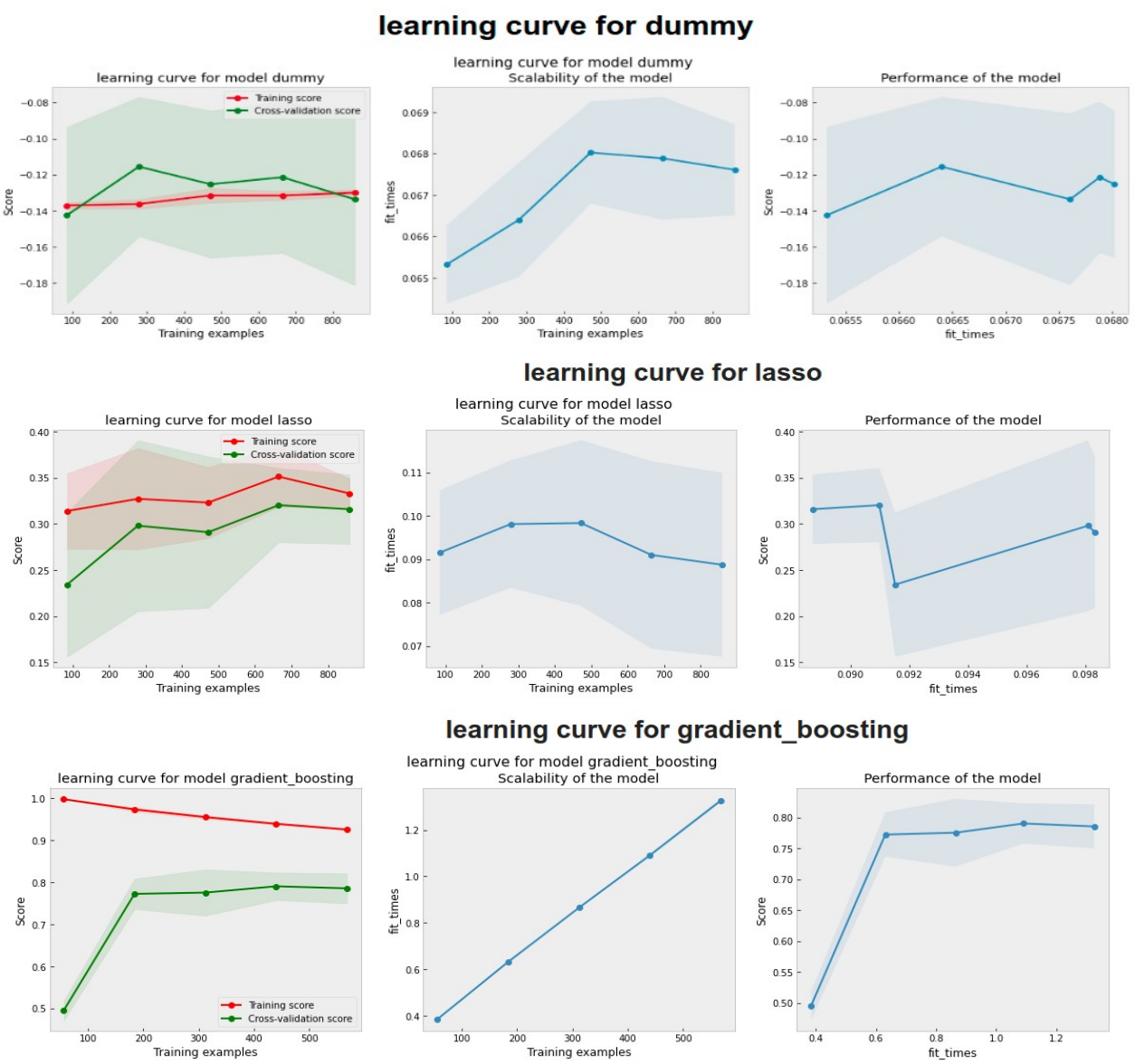
• Complexifier le modèle
• Ajouter plus de variables
• Laisser l'entrainement pendant plus de temps

Overfitting

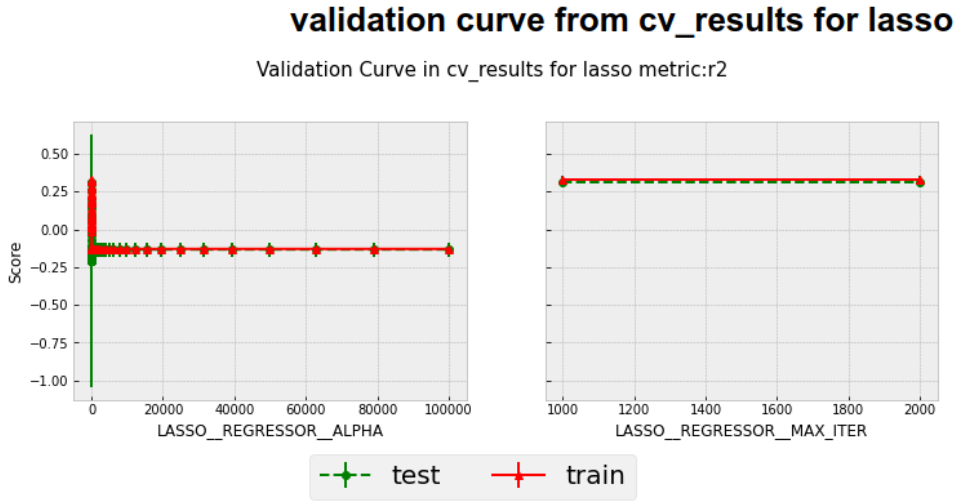
• Erreur d'entrainement très faible
• Erreur d'entrainement beaucoup plus faible que l'erreur de test
• Variance élevée

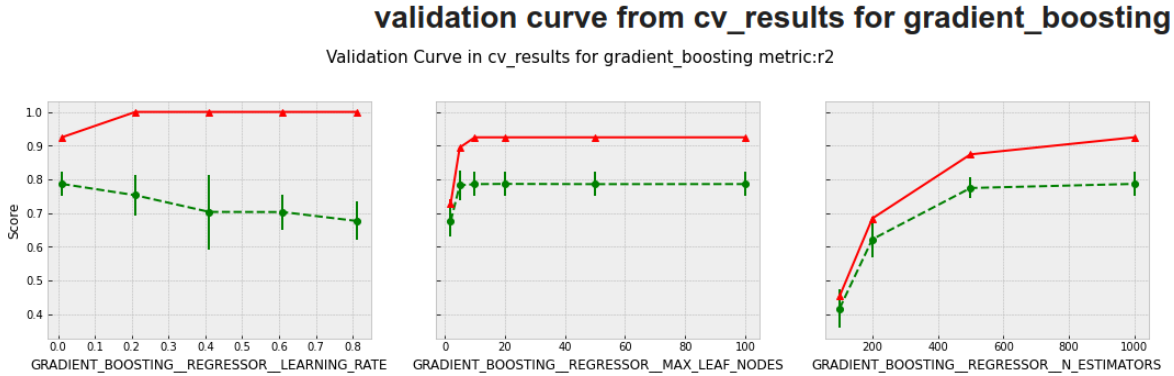
Remède possible

• Effectuer une régularisation
• Avoir plus de données



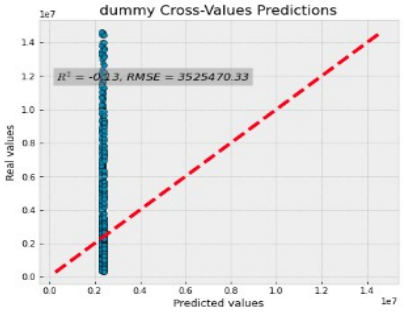
Résultat du tuning des hyperparamètres

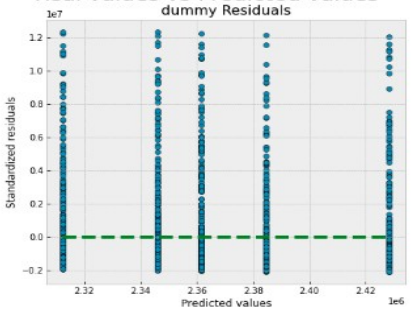


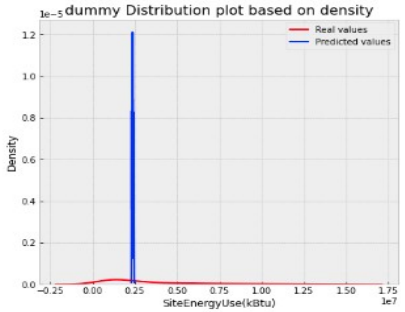


Résultat issus du gridSearchCV

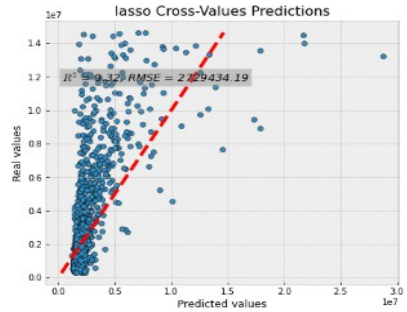
prediction performance for dummy

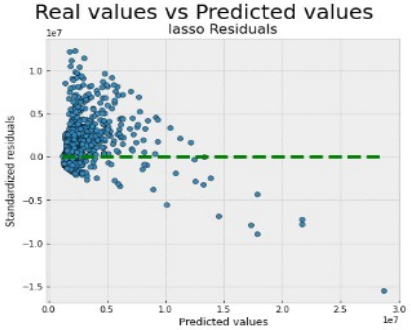


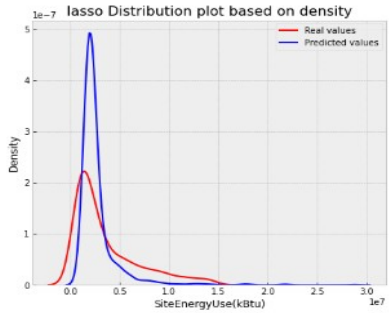




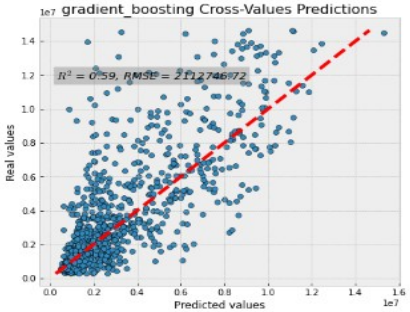
prediction performance for lasso

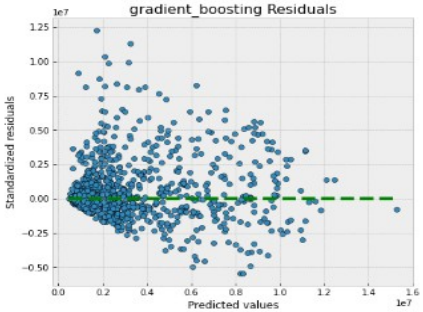


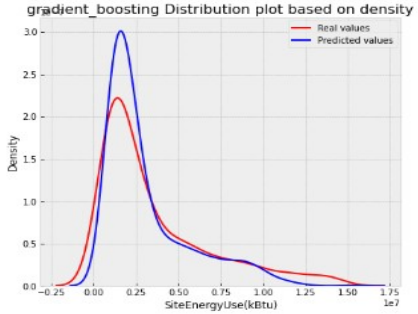




prediction performance for gradient_boosting







best parameter for dummy

	estimator	best_params
0	dummy	{'dummy__regressor__strategy': 'mean'}

best model score dummy

	estimator	R2	MAE	RMSE
0	dummy	0.134	2349435.923	3525321.572

best model score prediction for dummy

	estimator	R2	MAE	RMSE
0	dummy	-0.10038	2158287.124	3252122.254

best parameter for lasso

	estimator	best_params
4	lasso	{'lasso__regressor__alpha': 0.1747528400007683, 'lasso__regressor__max_iter': 1000}

best model score lasso

	estimator	R2	MAE	RMSE
4	lasso	0.316	1737910.221	2729449.726

best model score prediction for lasso

	estimator	R2	MAE	RMSE
4	lasso	-1.21317	1851654.659	4612137.04

best model score gradient_boosting

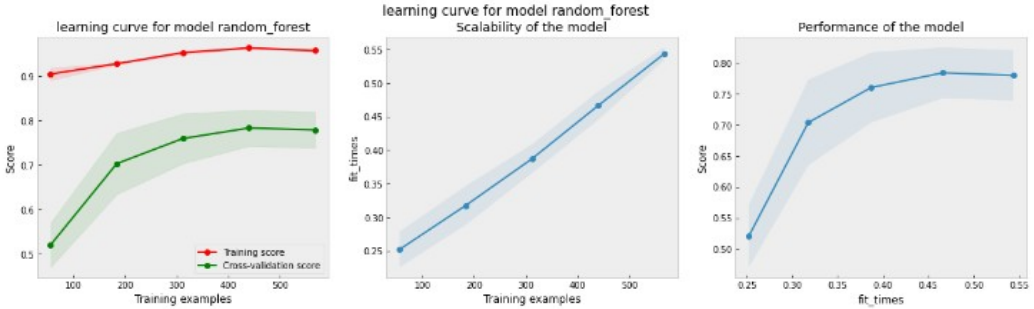
	estimator	R2	MAE	RMSE
8	gradient_boosting	0.593	1327553.101	2110907.31

best model score prediction for gradient_boosting

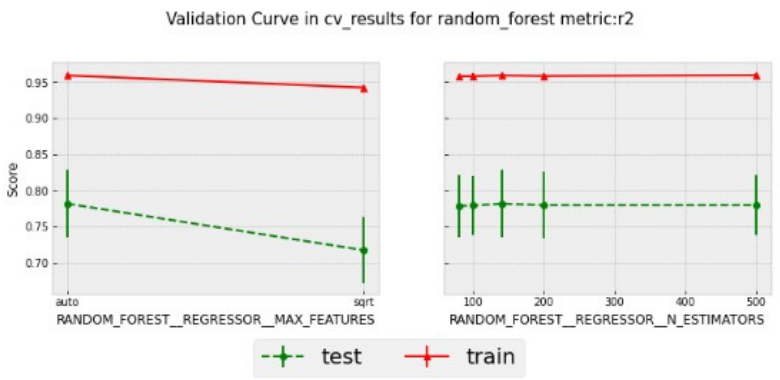
	estimator	R2	MAE	RMSE
8	gradient_boosting	0.66866	1143572.282	1784567.109

Résultat sur les meilleurs modèles de prédictions avec energy starscore

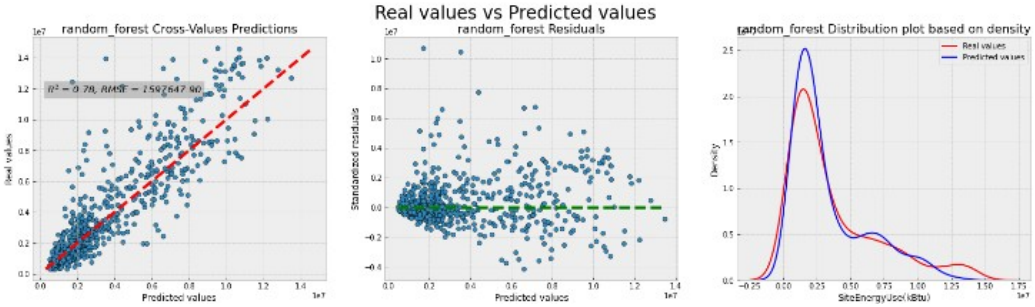
learning curve for random_forest



validation curve from cv_results for random_forest



prediction performance for random_forest



best parameter for random_forest

	estimator	best_params
6	random_forest	{'random_forest__regressor__max_features': 'auto', 'random_forest__regressor__n_estimators': 500}

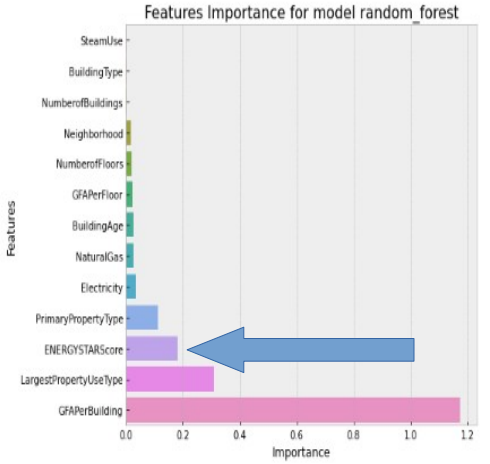
best model score random_forest

	estimator	R2	MAE	RMSE
6	random_forest	0.781	976113.299	1586096.343

best model score prediction for random_forest

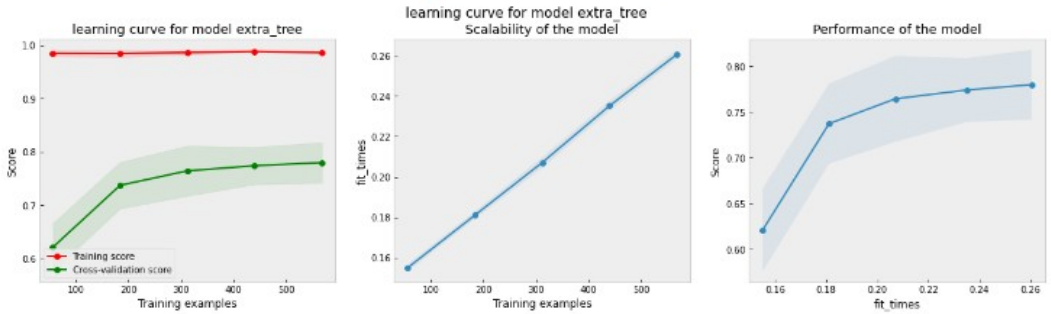
	estimator	R2	MAE	RMSE
6	random_forest	0.83478	788627.435	1227295.353

features importances for random_forest



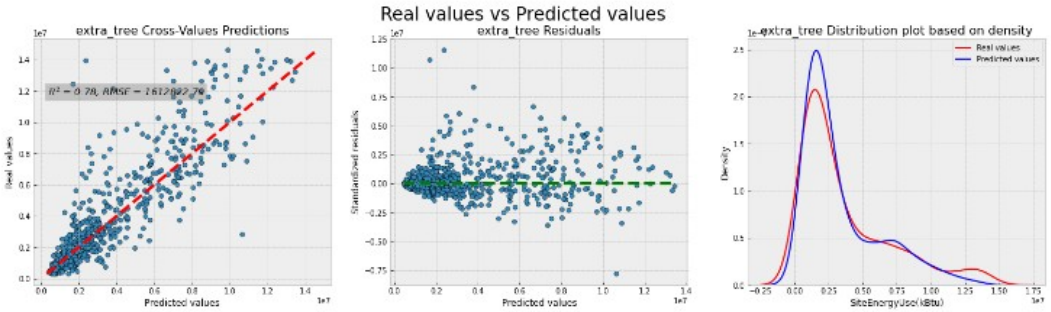
Résultat sur les meilleurs modèles de prédictions avec energy starscore

learning curve for extra_tree



validation curve from cv_results for extra_tree

prediction performance for extra_tree



best parameter for extra_tree

	estimator	best_params
7	extra_tree	{'extra_tree__regressor__bootstrap': False, 'extra_tree__regressor__min_samples_split': 4, 'extra_tree__regressor__n_estimators': 80}

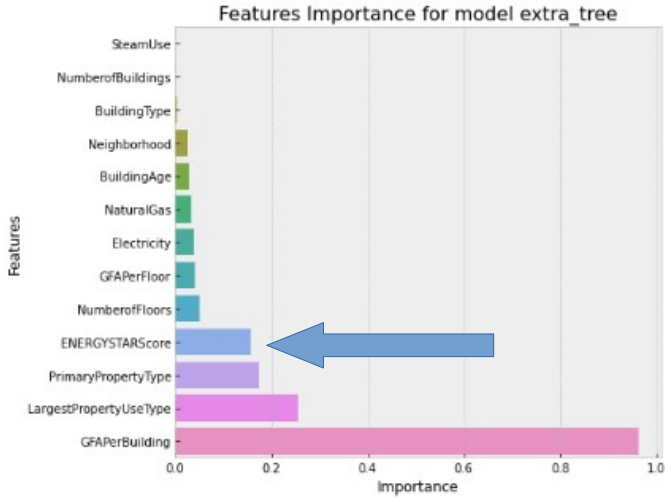
best model score extra_tree

	estimator	R2	MAE	RMSE
7	extra_tree	0.78	957827.013	1595224.387

best model score prediction for extra_tree

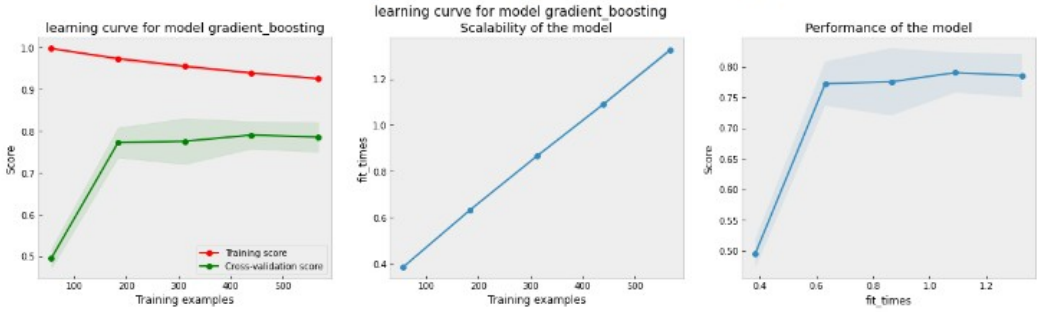
	estimator	R2	MAE	RMSE
7	extra_tree	0.82798	801659.332	1252270.23

features importances for extra_tree

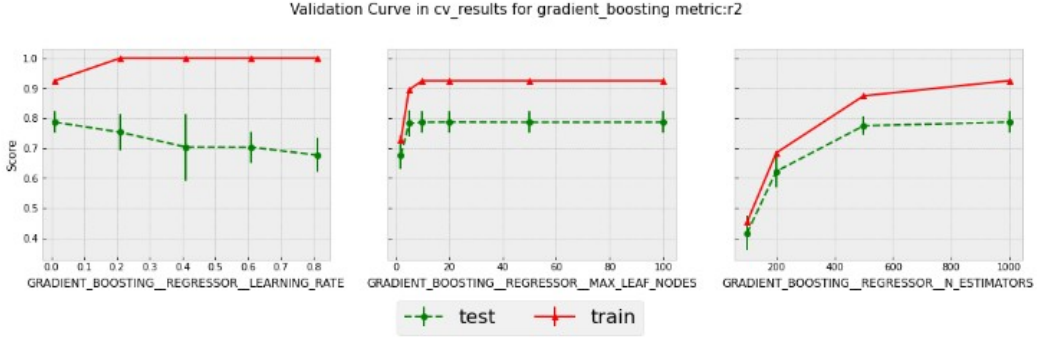


Résultat sur les meilleurs modèles de prédictions avec energy starscore

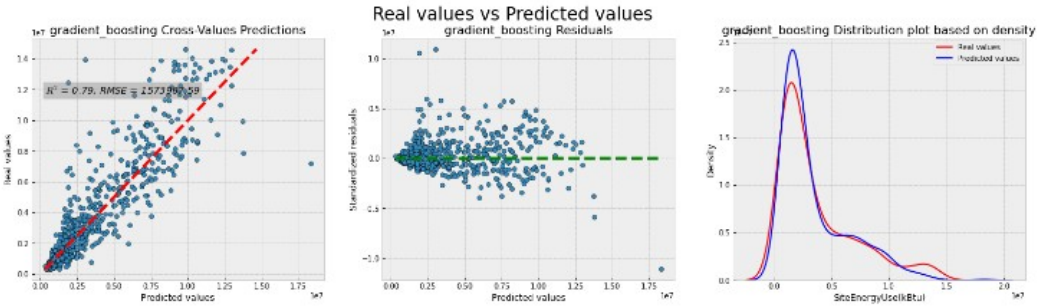
learning curve for gradient_boosting



validation curve from cv_results for gradient_boosting



prediction performance for gradient_boosting



best parameter for gradient_boosting

	estimator	best_params
8	gradient_boosting	{'gradient_boosting__regressor__learning_rate': 0.01, 'gradient_boosting__regressor__max_leaf_nodes': 10, 'gradient_boosting__regressor__n_estimators': 1000}

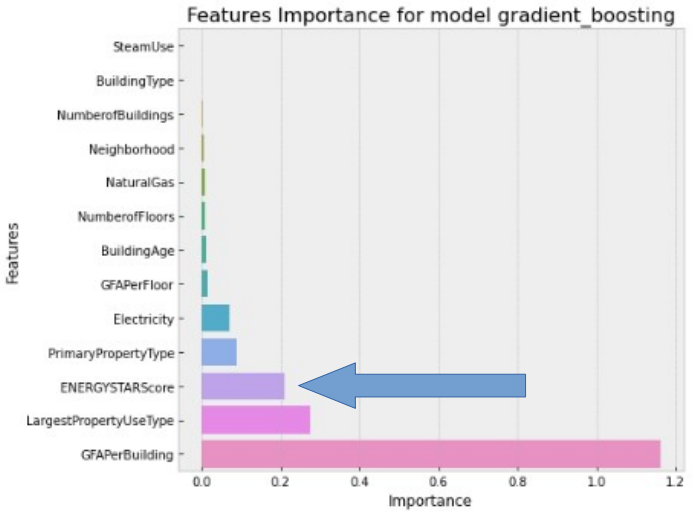
best model score gradient_boosting

	estimator	R2	MAE	RMSE
8	gradient_boosting	0.786	945925.639	1570978.692

best model score prediction for gradient_boosting

	estimator	R2	MAE	RMSE
8	gradient_boosting	0.85022	786564.119	1168534.413

features importances for gradient_boosting



Comparaisons avec et sans energy star score

Prédiction de l'énergie

estimator		R2	MAE	RMSE
dummy	options with energy	-0.08305	2088069.333	3.142226e+06
	options and no energy	-0.10038	2158287.124	3.252122e+06
linear	options with energy	-1.40542	1409614.634	4.682829e+06
	options and no energy	-46.34157	3053103.836	2.133123e+07
elastic_net	options with energy	0.33161	1613216.708	2.468477e+06
	options and no energy	-0.41428	1786815.168	3.686906e+06
ridge	options with energy	-1.10621	3146911.744	4.381911e+06
	options and no energy	-0.87808	1830047.140	4.248659e+06
lasso	options with energy	-1.13571	3208632.563	4.412490e+06
	options and no energy	-1.21317	1851654.659	4.612137e+06
svr	options with energy	0.81939	820478.924	1.283167e+06
	options and no energy	0.62650	1239677.193	1.894686e+06
random_forest	options with energy	0.83478	788627.435	1.227295e+06
	options and no energy	0.66592	1150759.898	1.791938e+06
extra_tree	options with energy	0.82798	801659.332	1.252270e+06
	options and no energy	0.66948	1154949.305	1.782357e+06
gradient_boosting	options with energy	0.85022	786564.119	1.168534e+06
	options and no energy	0.66817	1143908.897	1.785874e+06

Prédiction du CO2

estimator		R2	MAE	RMSE
dummy	options with energy	-0.10651	49.723	82.758
	options and no energy	-0.11684	52.130	85.501
linear	options with energy	0.52103	25.585	54.449
	options and no energy	-3.54234	48.053	172.432
elastic_net	options with energy	0.38426	34.016	61.735
	options and no energy	-0.27509	40.441	91.358
ridge	options with energy	-0.65543	66.038	101.225
	options and no energy	-3.73979	47.567	176.139
lasso	options with energy	-0.71840	67.602	103.133
	options and no energy	-0.14196	38.926	86.457
svr	options with energy	0.84968	18.399	30.503
	options and no energy	0.63124	28.203	49.130
random_forest	options with energy	0.79636	19.332	35.503
	options and no energy	0.64500	26.046	48.205
extra_tree	options with energy	0.80398	19.002	34.833
	options and no energy	0.61183	26.335	50.407
gradient_boosting	options with energy	0.86870	16.955	28.508
	options and no energy	0.66990	26.241	46.483

Conclusions

Les courbes d’apprentissage montrent qu’il est nécessaire d’avoir plus de données pour obtenir de meilleurs résultats sur les modèles, en effet en testant nos modèles avec les bâtiments résidentiel nos performances sont bien meilleurs par exemple

Il est nécessaire de bien apprendre le comportement de chaque modèle et ses paramètres pour en tirer le meilleur parti.

Les méthodes ensemblistes ont les meilleurs résultats en particulier Gradient Boosting

Il est recommandé de considérer EnergySTARScore dès le début

Questions/Réponses

Thank you!