



Seattle



Anticiper les besoins en
consommation de bâtiments

Sommaire

1. Contexte du projet

2. Présentation du jeu de données

3. Présentation du feature engineering

4. Analyse exploratoire

5. Modélisation - Prétraitement des données

6. Modélisation – Méthodologie

7. Modélisation – Les différents modèles et leurs scores

8. Modélisation – Choix du modèle

9. Intérêt de l'ENERGY STAR Score

10. Features importance

11. Conclusion

Contexte du projet

Problématique : Pour atteindre l'objectif de ville neutre en émissions de carbone en 2050, nous allons nous intéresser à la consommation et aux émissions des bâtiments non destinés à l'habitation.



Objectif : Prédire les émissions de CO₂ et la consommation totale d'énergie de bâtiments non destinés à l'habitation.



Missions :

- Nettoyer la base de données de relevés de la ville pour l'année 2016
- Réaliser une courte analyse exploratoire de ces données
- Utiliser ces données pour tester différents modèles de prédictions
- Evaluer l'intérêt de l'ENERGY STAR Score



Résultat attendu : Trouver le meilleur modèle en termes de performance afin de pouvoir prédire la consommation totale d'énergie et les émissions de CO₂.

Présentation du jeu de données

#	Column	Non-Null Count	Dtype
0	OSEBuildingID	3376	non-null int64
1	DataYear	3376	non-null int64
2	BuildingType	3376	non-null object
3	PrimaryPropertyType	3376	non-null object
4	PropertyName	3376	non-null object
5	Address	3376	non-null object
6	City	3376	non-null object
7	State	3376	non-null object
8	ZipCode	3360	non-null float64
9	TaxParcelIdentificationNumber	3376	non-null object
10	CouncilDistrictCode	3376	non-null int64
11	Neighborhood	3376	non-null object
12	Latitude	3376	non-null float64
13	Longitude	3376	non-null float64
14	YearBuilt	3376	non-null int64
15	NumberOfBuildings	3368	non-null float64
16	NumberOfFloors	3376	non-null int64
17	PropertyGFATotal	3376	non-null int64
18	PropertyGFAParking	3376	non-null int64
19	PropertyGFABuilding(s)	3376	non-null int64
20	ListofAllPropertyUseTypes	3367	non-null object
21	LargestPropertyUseType	3356	non-null object
22	LargestPropertyUseTypeGFA	3356	non-null float64
23	SecondLargestPropertyUseType	1679	non-null object
24	SecondLargestPropertyUseTypeGFA	1679	non-null float64
25	ThirdLargestPropertyUseType	596	non-null object
26	ThirdLargestPropertyUseTypeGFA	596	non-null float64
27	YearsENERGYSTARCertified	119	non-null object
28	ENERGYSTARScore	2533	non-null float64
29	SiteEUI(kBtu/sf)	3369	non-null float64
30	SiteEUIWN(kBtu/sf)	3370	non-null float64
31	SourceEUI(kBtu/sf)	3367	non-null float64
32	SourceEUIWN(kBtu/sf)	3367	non-null float64
33	SiteEnergyUse(kBtu)	3371	non-null float64
34	SiteEnergyUseWN(kBtu)	3370	non-null float64
35	SteamUse(kBtu)	3367	non-null float64
36	Electricity(kwh)	3367	non-null float64
37	Electricity(kBtu)	3367	non-null float64
38	NaturalGas(therms)	3367	non-null float64
39	NaturalGas(kBtu)	3367	non-null float64
40	DefaultData	3376	non-null bool
41	Comments	0	non-null float64
42	ComplianceStatus	3376	non-null object
43	Outlier	32	non-null object
44	TotalGHGEmissions	3367	non-null float64
45	GHGEmissionsIntensity	3367	non-null float64

Le jeu de données est composé de 3376 bâtiments et 46 variables.

Il comporte aussi bien des variables quantitatives que qualitatives.

Les bâtiments ne sont pas tous non destinés à l'habitation.

Certaines variables ne seront pas utiles pour les prédictions.

Présentation du feature engineering

Plusieurs modifications de variables ont été réalisées en plus de la réduction du nombre de variables :

- **EraBuild** : on reprend l'année de construction et on crée des époques pour avoir des intervalles de 10 ans.
- **SteamUse_Ratio, ElectricityUse_Ratio et NaturalGasUse_Ratio** : on reprend les relevés pour connaître la proportion d'utilisation de chacune des énergies.

Données finales :

#	Column	Non-Null Count	Dtype
0	PrimaryPropertyType	1475 non-null	object
1	Neighborhood	1475 non-null	object
2	NumberofBuildings	1475 non-null	float64
3	NumberofFloors	1475 non-null	int64
4	PropertyGFA_Total	1475 non-null	int64
5	PropertyGFA_Parking	1475 non-null	int64
6	LargestPropertyUseType	1475 non-null	object
7	LargestPropertyUseType_GFA	1475 non-null	float64
8	SecondLargestPropertyUseType	1475 non-null	object
9	SecondLargestPropertyUseType_GFA	1475 non-null	float64
10	ThirdLargestPropertyUseType	1475 non-null	object
11	ThirdLargestPropertyUseType_GFA	1475 non-null	float64
12	ENERGYSTARScore	960 non-null	float64
13	TARGET_SiteEnergyUse(kBtu)	1475 non-null	float64
14	TARGET_TotalGHGEmissions	1475 non-null	float64
15	EraBuild	1475 non-null	int64
16	SteamUse_Ratio	1475 non-null	float64
17	ElectricityUse_Ratio	1475 non-null	float64
18	NaturalGasUse_Ratio	1475 non-null	float64

Variables Quantitatives :

```
['NumberofBuildings',  
 'NumberofFloors',  
 'PropertyGFA_Total',  
 'PropertyGFA_Parking',  
 'LargestPropertyUseType_GFA',  
 'SecondLargestPropertyUseType_GFA',  
 'ThirdLargestPropertyUseType_GFA',  
 'ENERGYSTARScore',  
 'EraBuild',  
 'SteamUse_Ratio',  
 'ElectricityUse_Ratio',  
 'NaturalGasUse_Ratio']
```

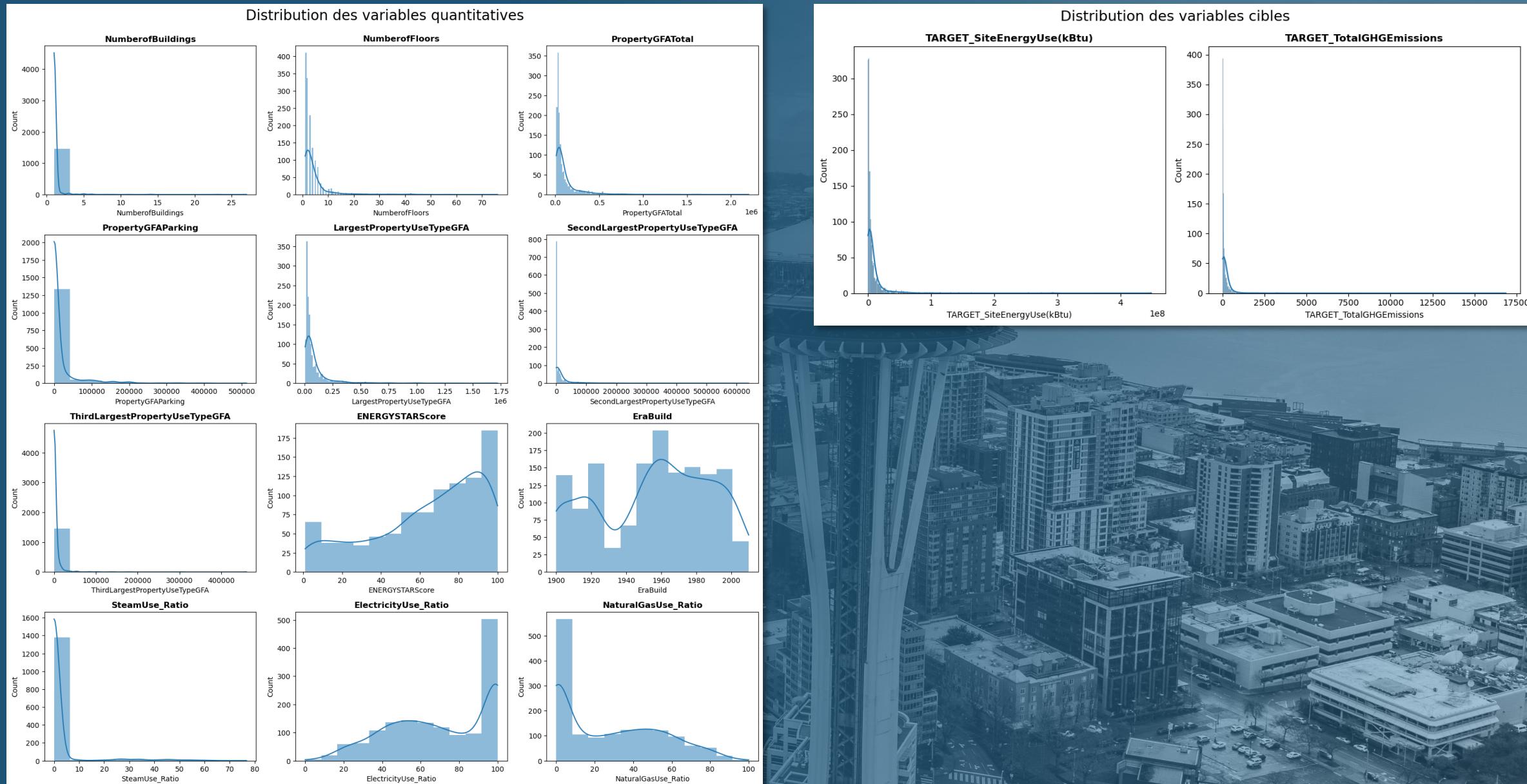
Variables Qualitatives :

```
['PrimaryPropertyType',  
 'Neighborhood',  
 'LargestPropertyUseType',  
 'SecondLargestPropertyUseType',  
 'ThirdLargestPropertyUseType']
```

Variables Cibles :

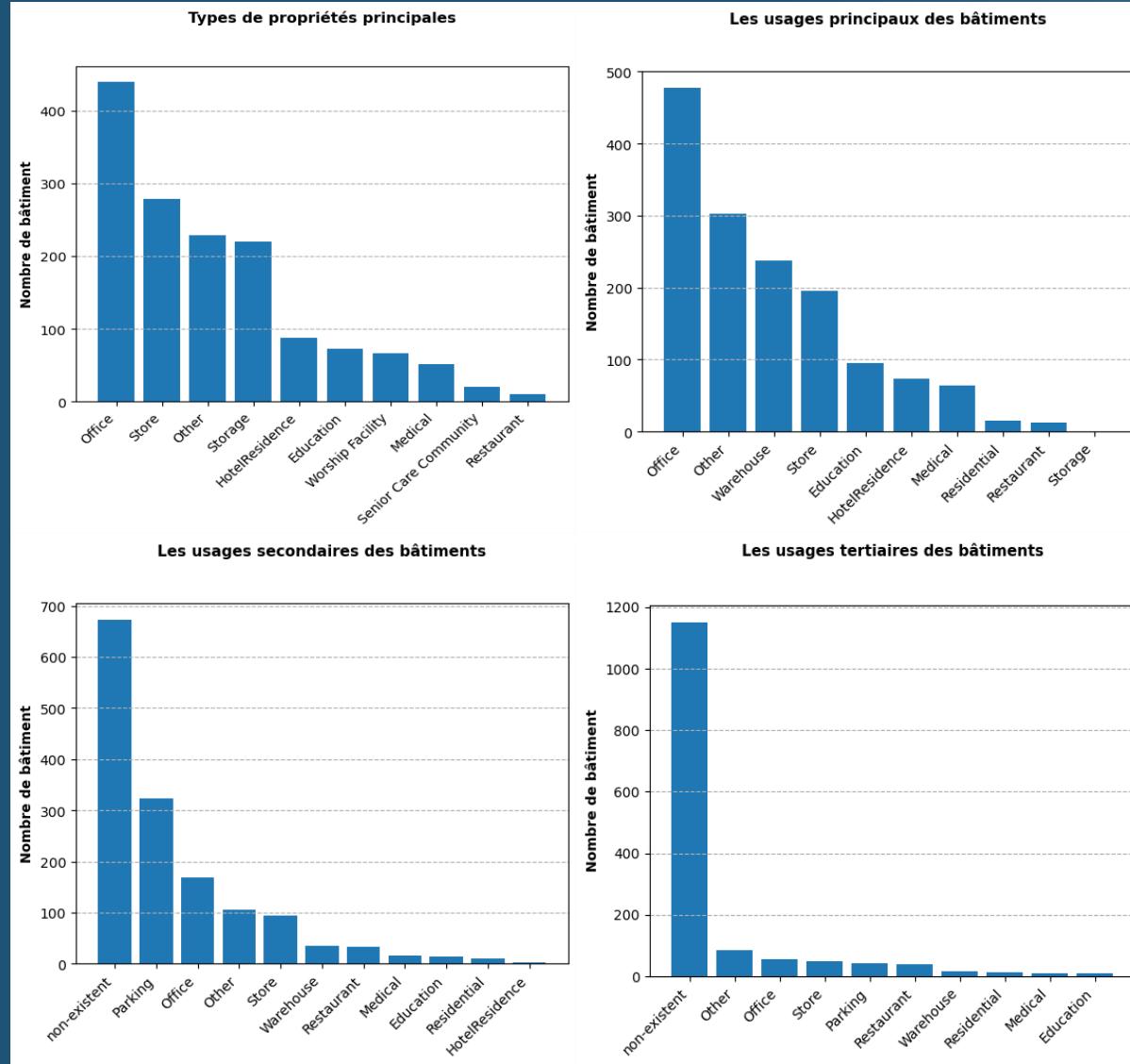
```
['TARGET_SiteEnergyUse(kBtu)', 'TARGET_TotalGHGEmissions']
```

Analyse exploratoire .1/3



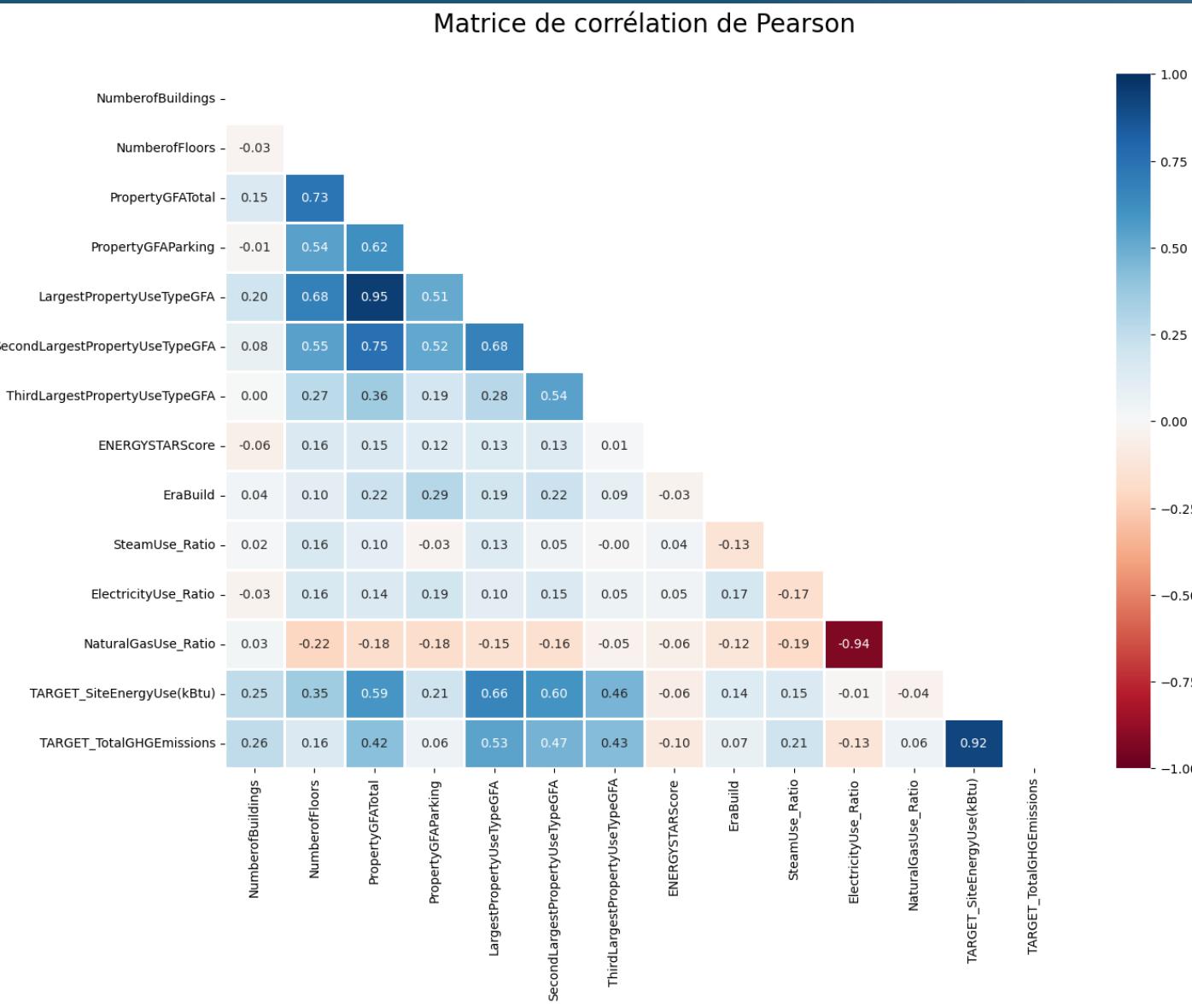
Analyse exploratoire .2/3

La distribution des proportions des variables catégorielles



Analyse exploratoire .3/3

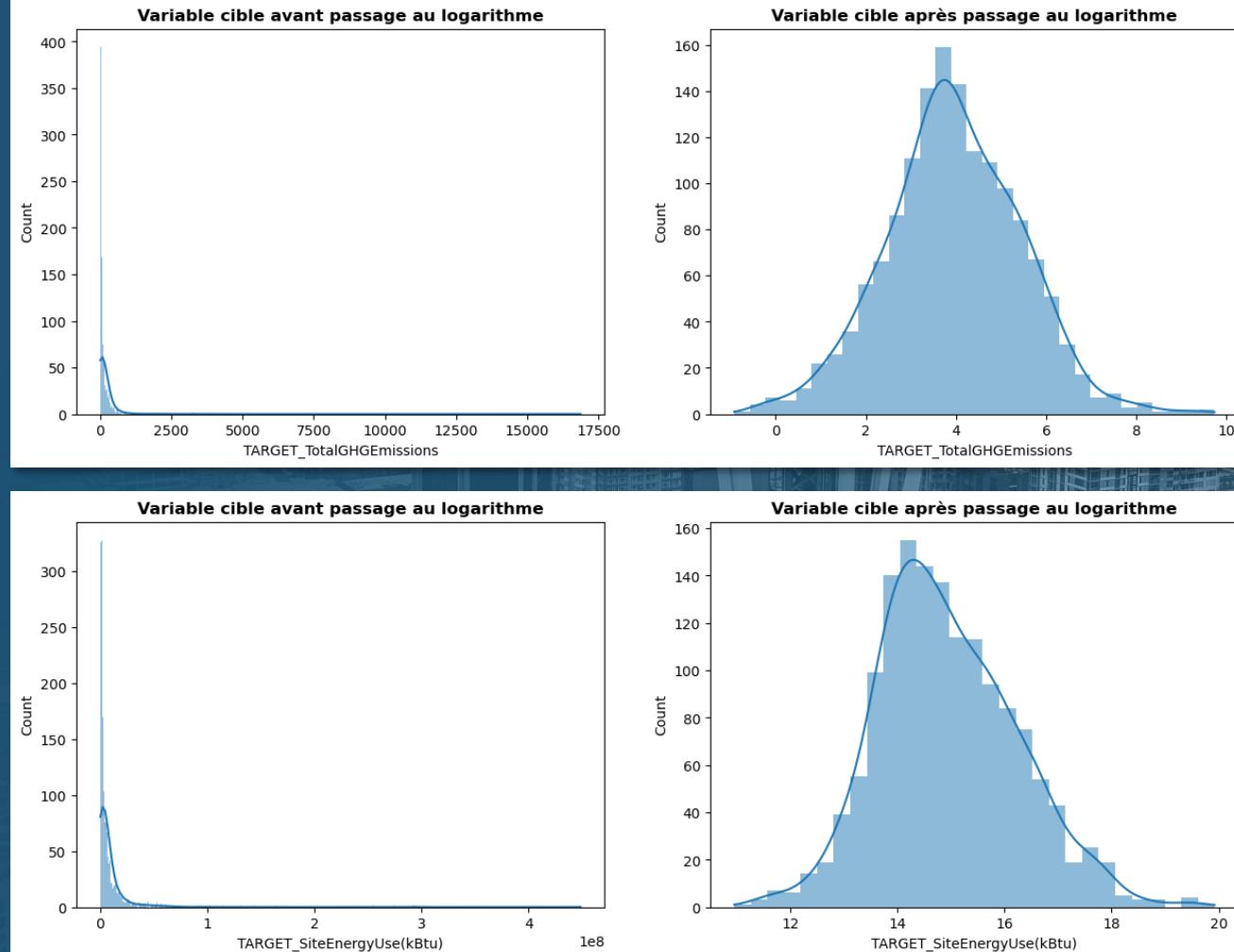
Matrice de corrélation de Pearson



- Les variables cibles sont très corrélées entre elles.
- Les autres variables les plus corrélées entre elles sont les variables de surfaces (GFA).
- Et, les variables de ratios d'utilisation d'électricité et de gaz sont anti-corrélées.

Modélisation - Prétraitement des données .1/2

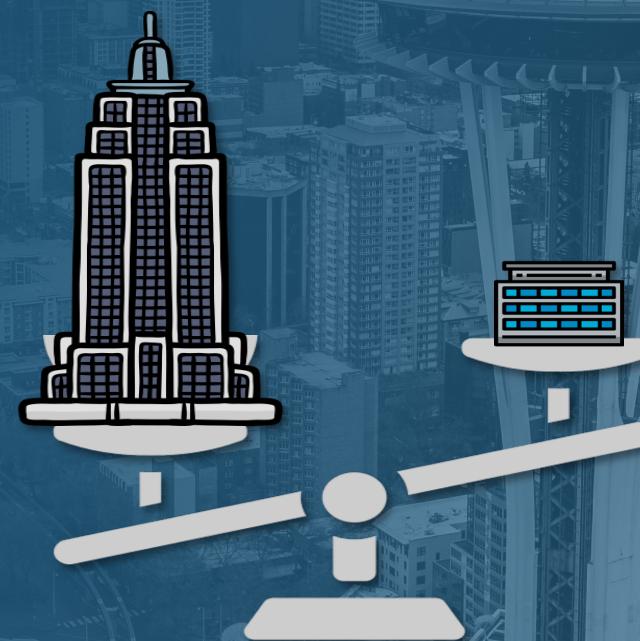
La distribution des variables cibles est étalée vers la droite, nous pourrions « normaliser » la distribution en passant ces variables au logarithme.



Modélisation - Prétraitement des données .2/2

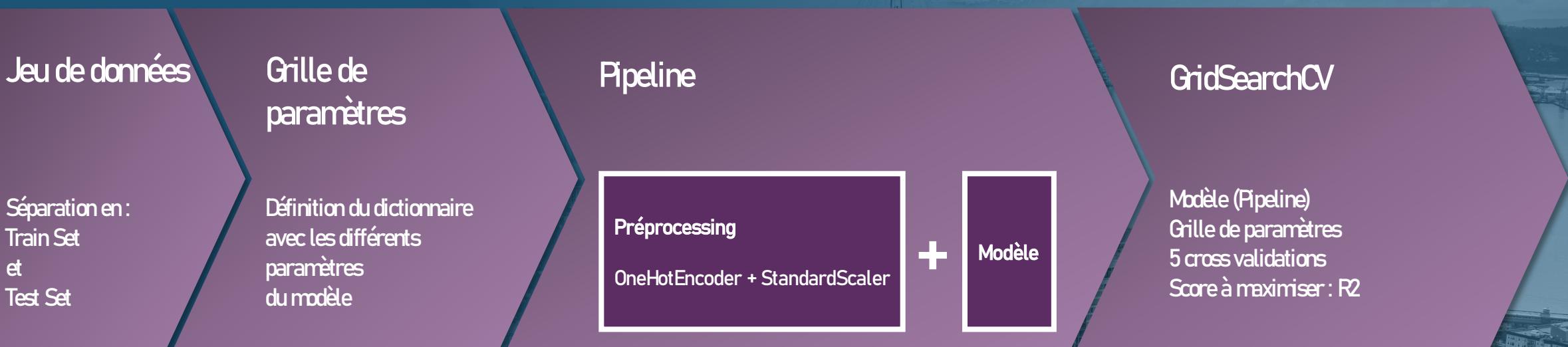
Les **variables qualitatives** ont été **encodées** avec un OneHotEncoder afin de pouvoir créer une matrice avec pour chaque valeurs de ces variables, une nouvelles variables avec des valeurs booléennes.

Les **variables quantitatives** ont été **standardisées** avec un StandardScaler afin de rendre les variables indépendantes de leur unité ou de leur échelle d'origine.



Modélisation – Méthodologie

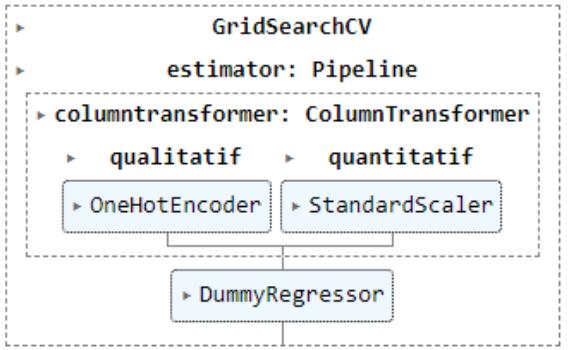
La méthodologie choisie consiste à entraîner différents modèles sur une partie des données (Train Set) et d'utiliser la cross validation pour optimiser les différents paramètres de chacun d'eux afin d'obtenir le meilleur score.



Fonction d'automatisation de la pipeline et de la grid search :

```
def final_pipeline(X, model, params):  
    variables_quantitatives = X.select_dtypes(['int64', 'float64']).columns.to_list()  
    variables_qualitatives = X.select_dtypes(['object', 'bool']).columns.to_list()  
    quantitatif_scaler = StandardScaler()  
    qualitatif_encoder = OneHotEncoder(handle_unknown='ignore')  
    transformers = [('qualitatif', qualitatif_encoder, variables_qualitatives),  
                   ('quantitatif', quantitatif_scaler, variables_quantitatives)]  
    preparation = ColumnTransformer(transformers)  
    model_pipeline = make_pipeline(preparation, model)  
    grid = GridSearchCV(model_pipeline,  
                        params,  
                        cv=5,  
                        scoring='r2',  
                        n_jobs=-1)
```

Résultat avec le modèle DummyRegressor :



Modélisation – Les différents modèles et leurs scores .1/2

Plusieurs types de modèles ont été utilisés :

1) Modèle de référence

- *Dummy Regressor*: évaluer la performance des autres modèles

2) Modèles linéaires ou basés sur une forme de régression

- *Régression Linéaire*
- *Regression Ridge*: contraindre la régression linéaire en réduisant la variance des coefficient de régression
- *Lasso*: contraindre la régression linéaire en mettant certaines valeurs de coefficients de régression à 0
- *Elastic Net*: Ridge + Lasso
- *Linear SVR*: trouver la meilleure ligne qui servira de frontière de décision pour séparer au mieux les points

3) Modèles ensemblistes

- *Forêts aléatoires*: capter des relations non linéaires et gérer des ensembles de données complexes
- *XGBoost Regressor*: Gradient Boosting, créer une prédiction précise en combinant plusieurs modèles moins précis

Modélisation – Les différents modèles et leurs scores .2/2

Résultats pour la prédiction d'émission de CO2 :

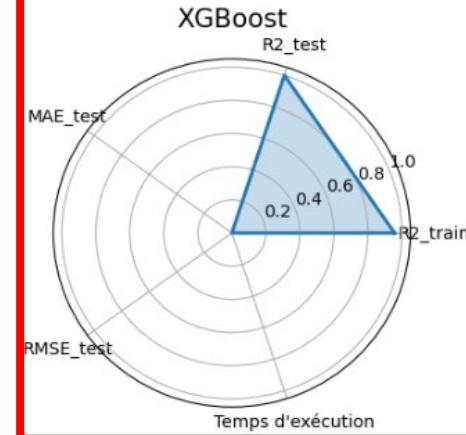
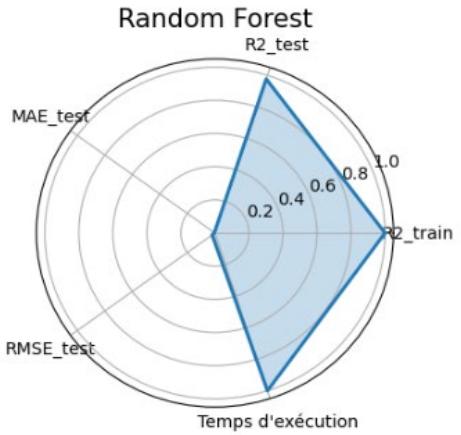
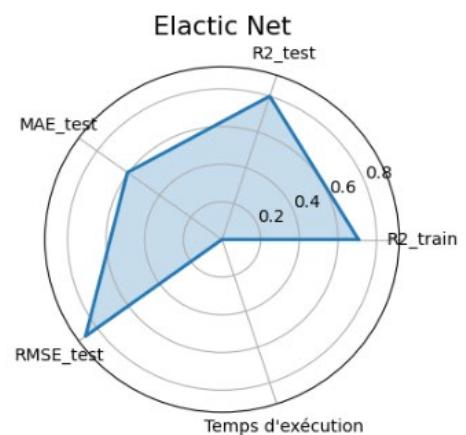
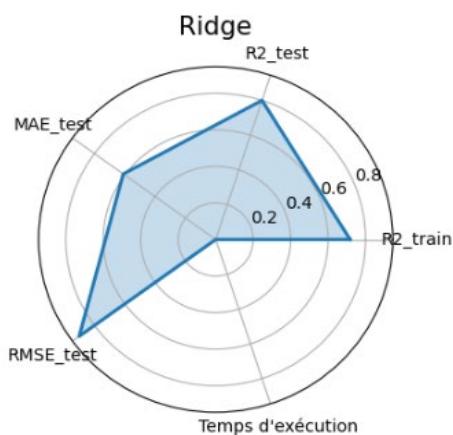
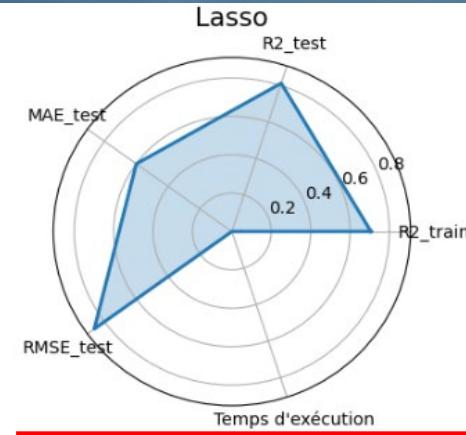
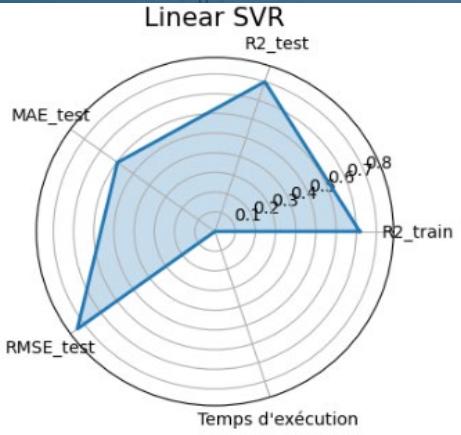
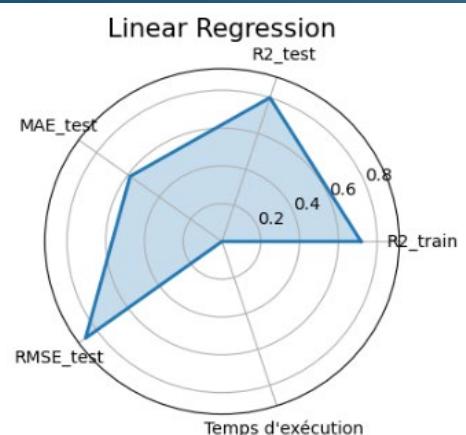
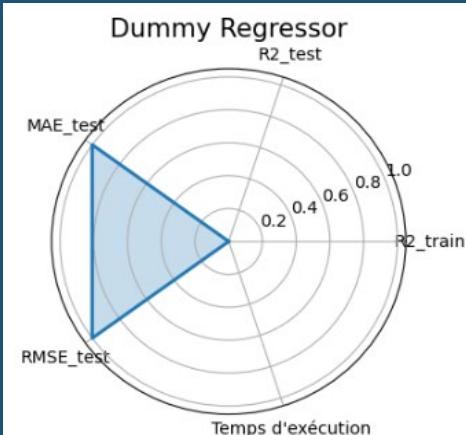
	Model	R2_train	R2_test	MAE_test	RMSE_test	Temps d'exécution
0	Dummy Regressor	-0.00	-0.01	188.14	600.15	0.003000
1	Linear Regression	0.66	0.67	152.00	563.46	0.003001
2	Ridge	0.66	0.67	154.07	571.87	0.003001
4	Elastic Net	0.65	0.67	153.69	564.34	0.003001
5	Linear SVR	0.66	0.67	152.88	555.39	0.003001
3	Lasso	0.65	0.68	153.17	561.47	0.003001
6	Random Forest	0.92	0.82	101.12	322.07	0.066015
7	XGBoost	0.88	0.84	100.79	316.50	0.003001

Résultats pour la prédiction de consommation totale d'énergie :

	Model	R2_train	R2_test	MAE_test	RMSE_test	Temps d'exécution
0	Dummy Regressor	0.00	-0.01	7568317.25	18210191.08	0.003000
1	Linear Regression	0.58	0.57	7453760.28	38982612.56	0.002000
2	Ridge	0.57	0.57	7628227.19	40952351.31	0.002001
4	Elastic Net	0.56	0.57	7553267.26	40077875.31	0.003001
5	Linear SVR	0.58	0.57	7399879.18	37239503.20	0.003001
3	Lasso	0.56	0.58	7536817.53	39975143.61	0.002001
6	Random Forest	0.89	0.77	3630978.98	9138921.68	0.010002
7	XGBoost	0.83	0.78	4003271.85	9925096.99	0.003000

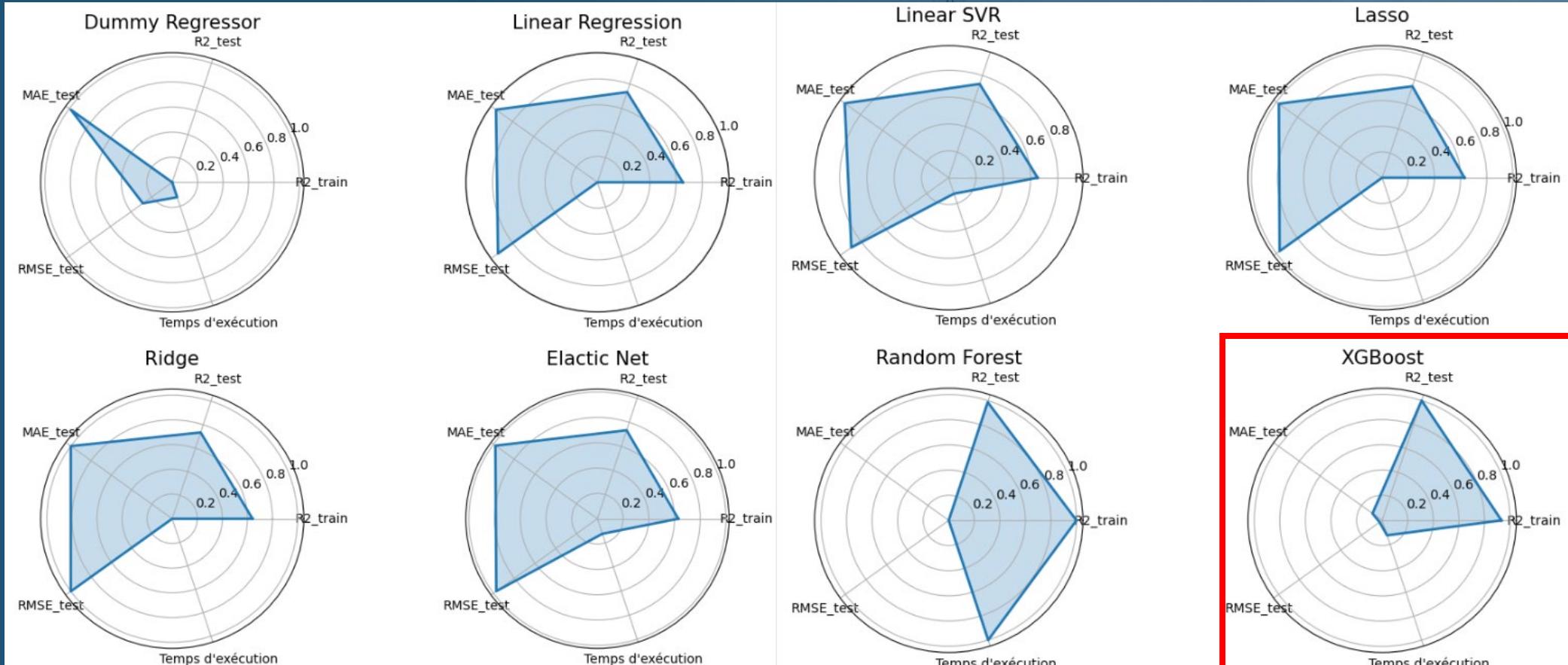
Modélisation – Choix du modèle .1/3

Pour la prédiction d'émission de CO2 :



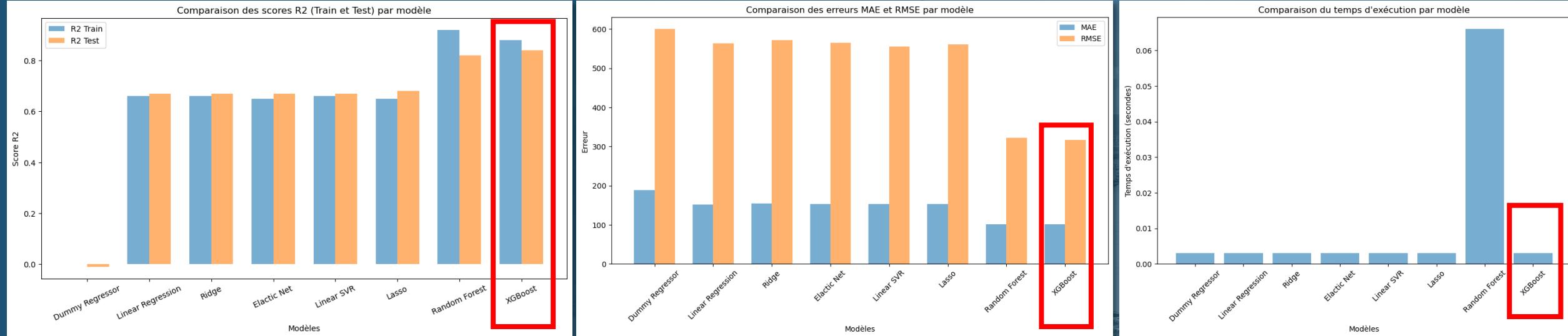
Modélisation – Choix du modèle .2/3

Pour la prédiction de consommation totale d'énergie :

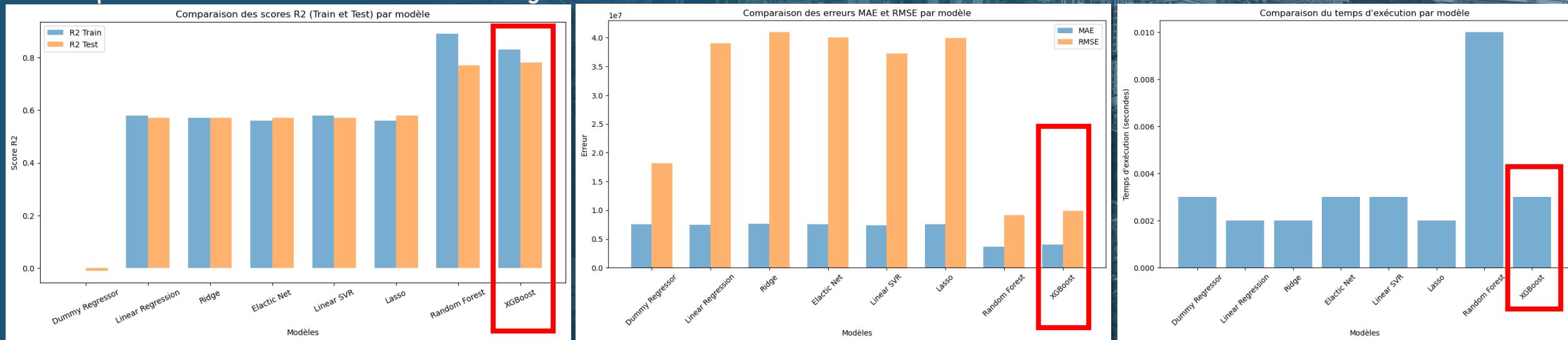


Modélisation – Choix du modèle .3/3

Pour la prédiction d'émission de CO2 :



Pour la prédiction de consommation totale d'énergie :



Intérêt de l'ENERGY STAR Score

Pour évaluer l'intérêt de l'ENERGY STAR Score, nous avons comparé les performances du modèle XGBoost que nous avons choisi en rajoutant la variable ENERGY STAR Score.

Résultats pour la prédiction d'émission de CO2 :

	Model	R2_train	R2_test	MAE_test	RMSE_test	Temps d'exécution
7	XGBoost	0.88	0.84	100.79	316.50	0.003001
8	XGBoost_ENERGY	0.98	0.86	67.84	345.58	0.004001



Résultats pour la prédiction de consommation totale d'énergie :

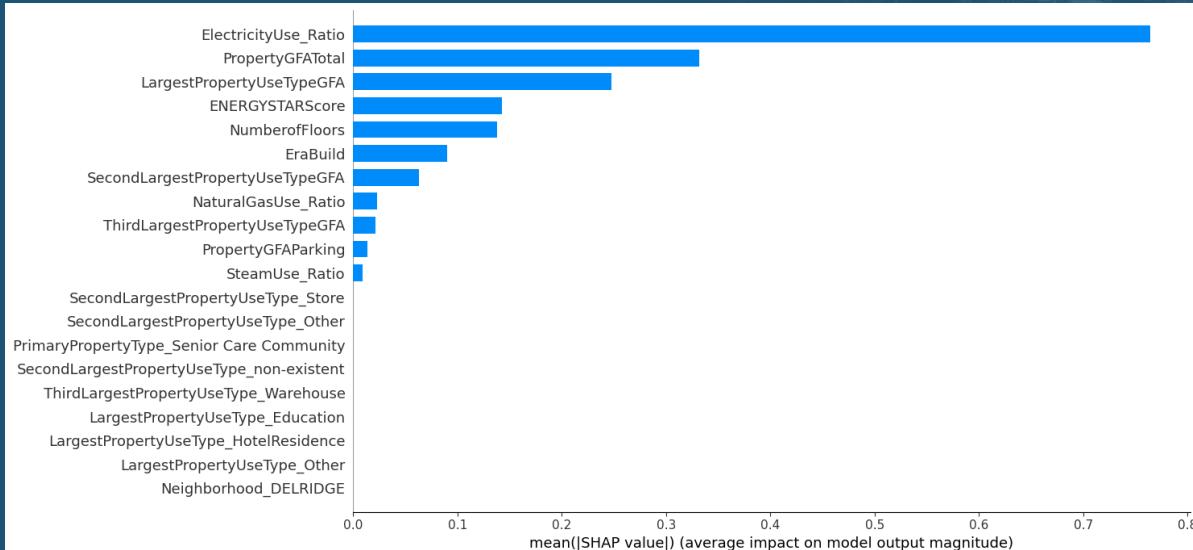
	Model	R2_train	R2_test	MAE_test	RMSE_test	Temps d'exécution
7	XGBoost	0.83	0.78	4003271.85	9925096.99	0.003000
8	XGBoost_ENERGY	0.96	0.82	4030301.56	28132278.22	0.004001

Pour prédire l'émission de CO2 ou la consommation totale d'énergie, l'ENERGY STAR Score permet d'améliorer les performances. Le choix de le garder pour maximiser le score du modèle semble judicieux.

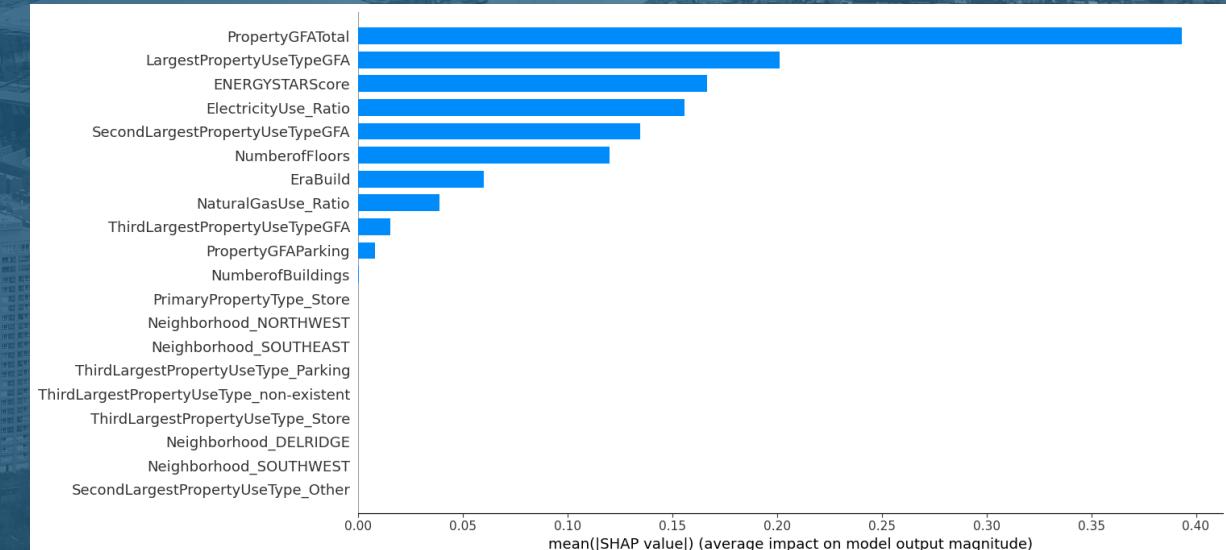
Features importance .1/2

Quelles sont les variables qui impact le plus le modèle ?

Pour la prédiction d'émission de CO₂ :



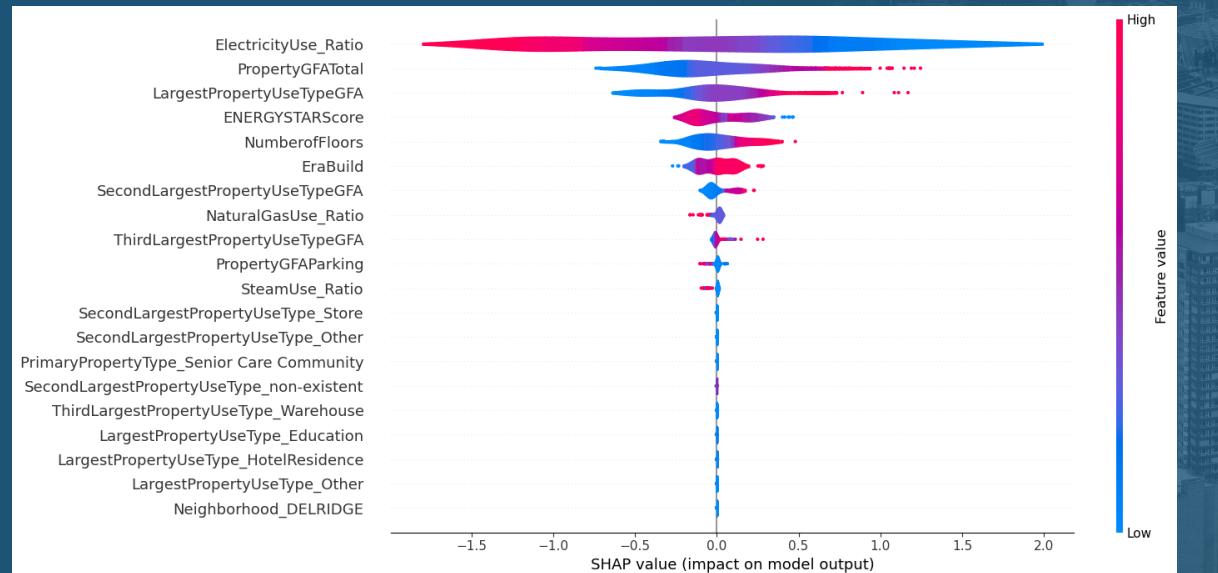
Résultats pour la prédiction de consommation totale d'énergie :



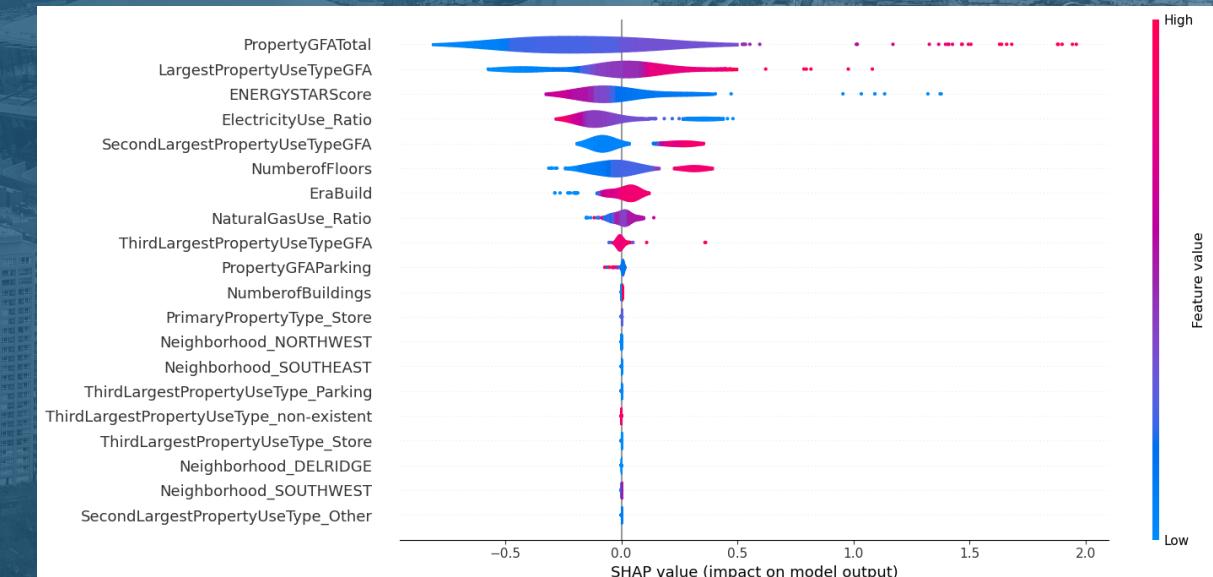
Features importance .1/2

De quelle manière ces variables impactent-elles le modèle ?

Pour la prédiction d'émission de CO₂ :



Résultats pour la prédiction de consommation totale d'énergie :



Conclusion

Le modèle XGBoost Regressor semble être le plus approprié pour prédire les données.

Le modèle en question nous permet de maximiser le score et minimiser l'erreur.

En plus de sa performance, le modèle XGBoost se distingue par sa rapidité d'exécution, ce qui le rend pratique pour des futures applications en temps réel ou des futures données volumineuses.

L'ENERGY STAR Score permet d'améliorer la qualité des prédictions.

Cependant, en raison de la complexité du calcul de l'ENERGY STAR Score, il serait judicieux de solliciter des conseils métiers pour évaluer si cette amélioration justifie des efforts supplémentaires.