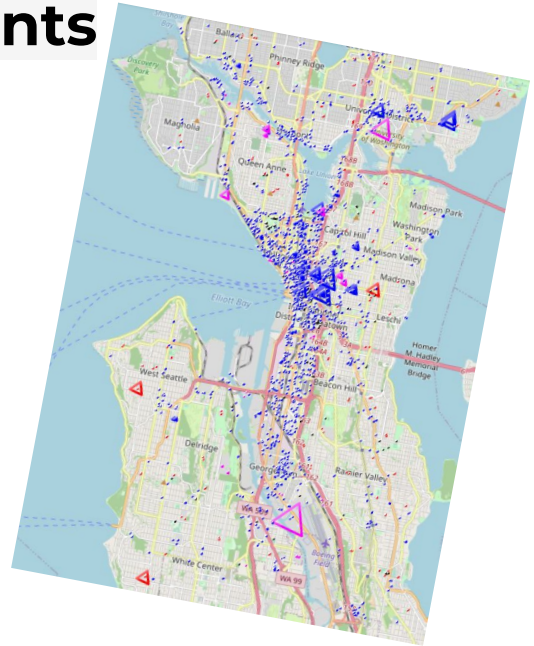
Anticipation des besoins en consommation électrique de bâtiments



Projet 3 parcours IML OpenClassrooms

Mai 2022

OPENCLASSROOMS

Plan

1. Problématique et présentation
2. Prétraitement des données
3. Modélisation et optimisation
4. Modèle final
5. Influence de L'ENERGYSTARScore
6. Conclusion

Problématique

Objectif de Seattle: Ville neutre en émission des Co_2

- Prédiction des émissions des Co_2
- Prédiction de la consommation de l'énergie totale

A partir des données existantes

- ❖ Relevés manuels minutieux effectuée en 2015 et 2016
- ❖ Intérêt de la feature ENERGYSTARTScore pour les prédictions

Présentation du dataset

Relevés 2015
3340 lignes
47 colonnes

Relevés 2016
3376 lignes
46 colonnes

- 1 ligne = 1 bâtiment
- Futures:Énergétique, géographique, surfacique, exploitation.

Concaténation des deux relevés

- Gestion des Nan pour les 2 relevés (85% taux de remplissage)
- Renommer et adaptation des colonnes

Prétraitement des données

Cleaning
Feature engineering
Exploration

Cleaning

- Concaténation des deux relevés
- Filtrage suivant les bâtiments non destinés à l'habitation

❖ Gestion des Nans

➢ Suppression des colonnes avec beaucoup de NaN (85%)

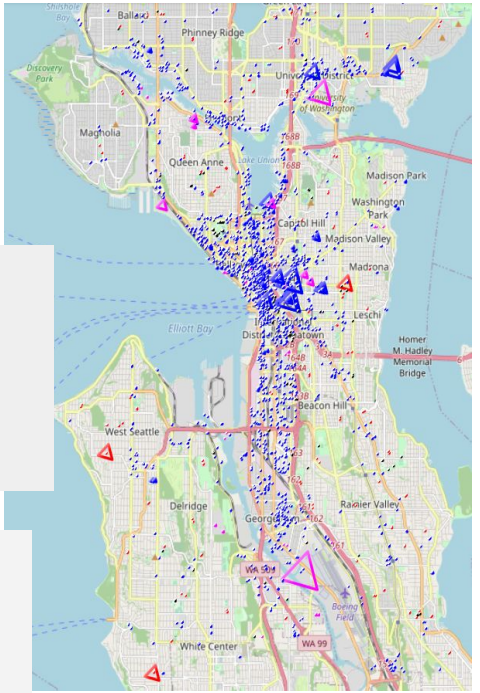
➢ Complétion des valeurs manquantes de catégories

➢ Imputation des valeurs numériques (imputation simple + KNN)

❖ Gestion des Outliers

➢ Valeurs négatives (remplacées par leurs modules)

➢ Valeurs extrêmes supprimées si elles sont fausses



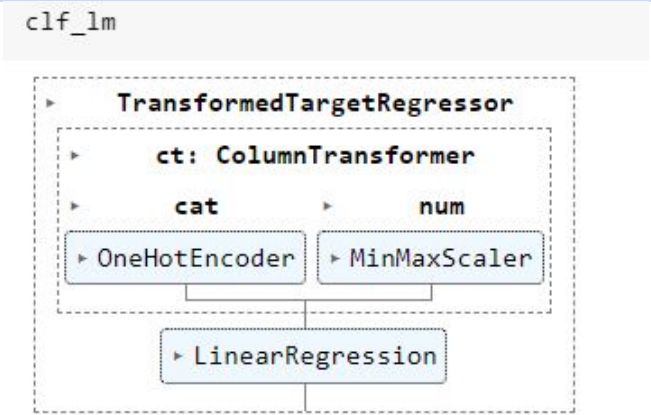
Feature engineering

- Suppression des features d'intensité de l'énergie (données coûteuses)
- Suppression des données de capteurs
- Utilisation du EnergyStarScore (puis supprimée ultérieurement)
- Suppression des features fortement corrélées
- Suppression de colonnes non pertinentes (comment, DefaultData, propertyName, State, City, ZipCodes)
- Features de catégories facile à exploiter
- Données Géographiques
- Log(1+x)-transformation des variables à prédire

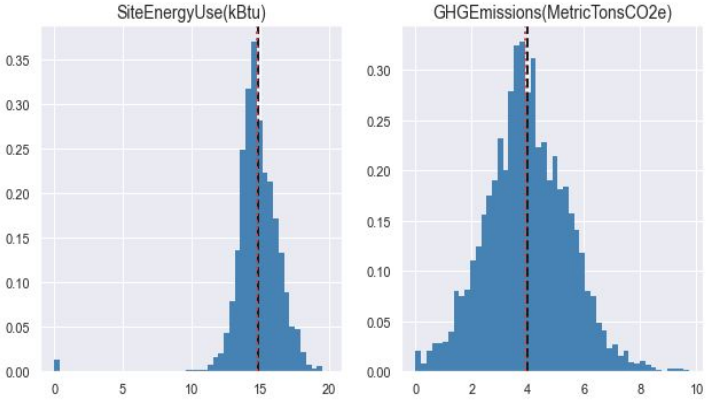
Imputation simple + KNN imputation des Nan

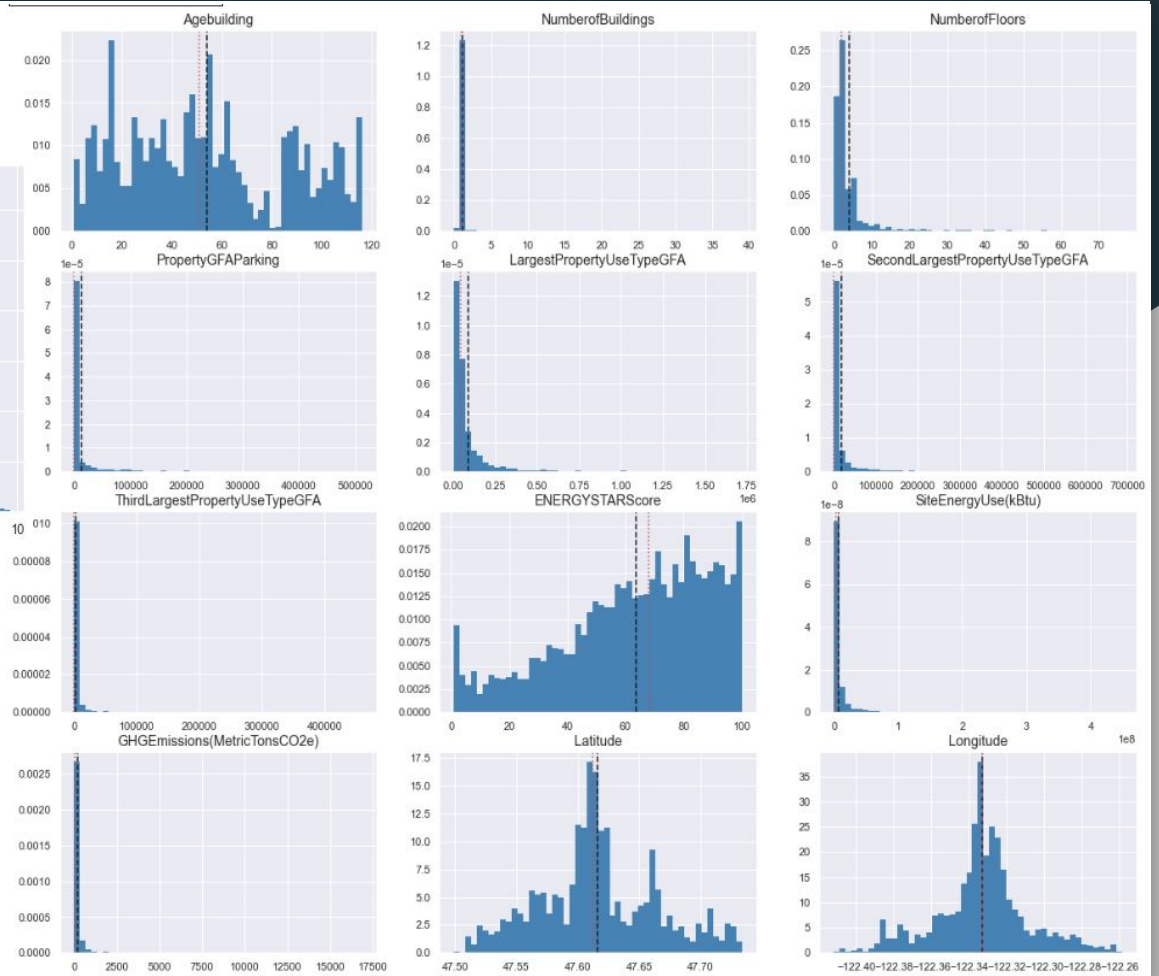
Variables catégorielles: OneHotEncoder

Variables numériques: MinMaxScaler

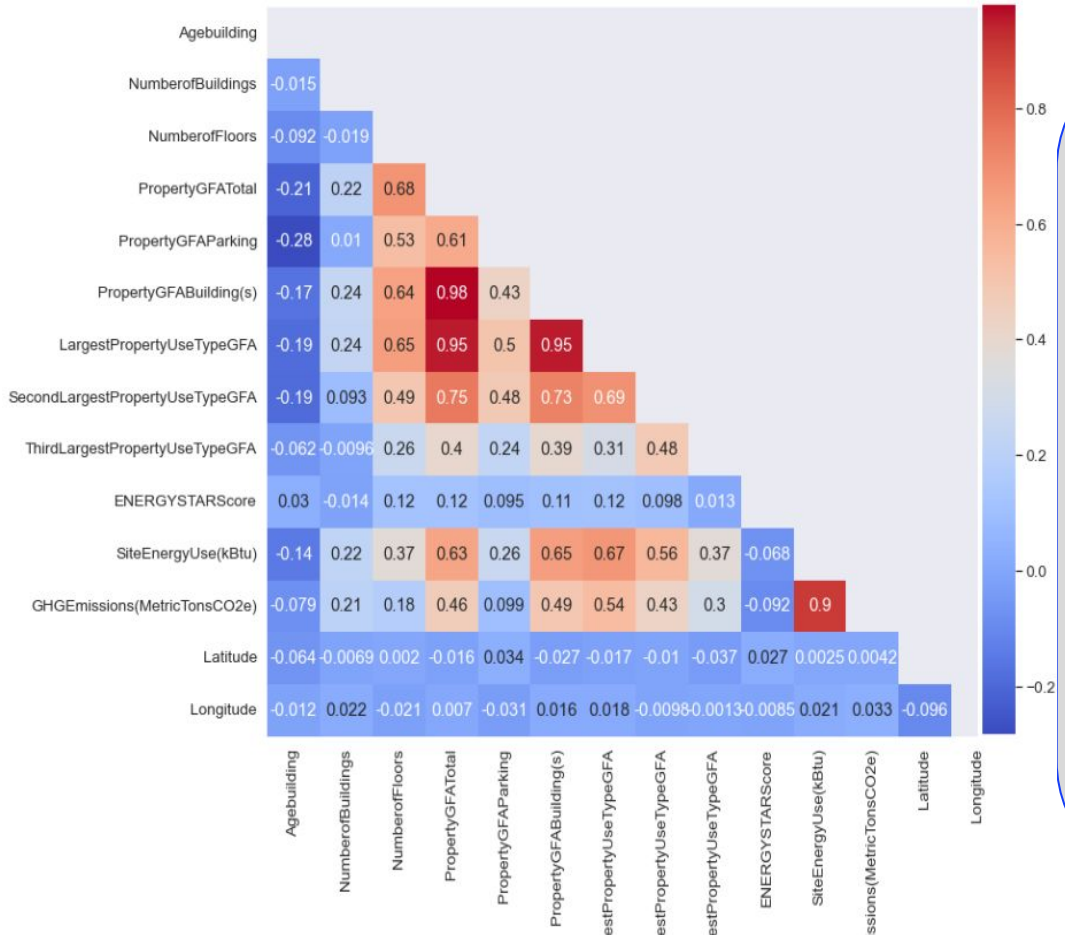


Exploration





- Distribution asymétrique (sauf Longitude)
- la transformation log(1+x) rend les variables à prédire gaussiennes



Corrélation importante entre les features de surface (typeGFA)

Suppression de “PropertyGFATotal” et “PropertyGFABuiding(s)”.

Corrélation importantes de variables cibles avec les features de surface

Pas de corrélation avec “ENERGYSTARScore”

Absence de corrélation négatives

Modélisation et optimisation

Modélisation

❖ Modèle linéaire:

- Régression Linéaire classique (MLR)
- Régression Ridge
- Régression Lasso
- Régression ElasticNet

❖ Modèle non-linéaire:

- Support vector Regression (SVR)
- Random Forest Regression (RFR)
- GradientBoostingRegressor (GBR)

	GHGEmissions(MetricTonsCO2e)	SiteEnergyUse(kBtu)
count	3213.000000	3.213000e+03
mean	173.608193	7.806362e+06
std	643.541305	2.045373e+07
min	0.000000	0.000000e+00
25%	19.810000	1.227772e+06
50%	49.180000	2.512319e+06
75%	138.960000	6.863561e+06
max	16870.980000	4.483853e+08

Métrique choisie: RMSE et R_2

Etapes de comparaison

Préparation des données training/test
Pipeline de transformation

- Grille de paramètres
- entraînement et optimisation des paramètres et des scores Avec **GridSearchCV**

Comparaison des modèles optimisés sur la base des scores RMSE, R_2, time-fit

Exemple de grille de paramètres:

Ridge	RFR	GBR
`alpha : np.linspace(0.1,40,41)`	`n_estimators : [10, 100, 300]`	`n_estimators : [10, 100, 500]`
`tol : [0.5,0.1,0.03,0.01]`	`min_samples_leaf : [1,2, 3, 4]`	`subsample : [0.8, 0.5, 0.1]` `max_depth : [6,8,10]`
	`max_features: ['auto', 'sqrt']`	`learning_rate : [0.02, 0.05]`

Résultats avec GridSearchCV

GHGEmissions

	R2	RMSE	MFT_total(s)	MFT(s)
Dummy	-0.0362	570.437	-	0.023
MLR	-845.51	7729.42	2.234	0.112
Ridge	0.489	344.834	42.636	0.035
Lasso	-0.061	604.74	79.69	0.033
El_Net	-0.05	602.606	177.352	0.042
SVR	0.516	360.005	275.948	0.767
RFR	0.662044	332.149251	1010.347	5.613
GBR	0.752	274.715	945.001	2.625

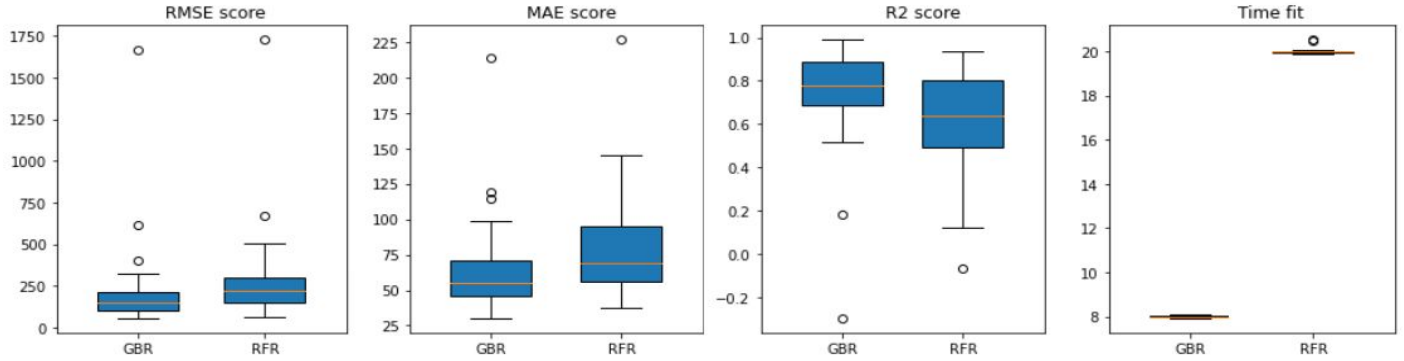
SiteEnergyUse

	R2	RMSE	MFT_total(s)	MFT(s)
Dummy	-0.0596	19625997.1003	-	0.021
MLR	-899.75	403466958.44	2.214	0.111
Ridge	0.5	12649539.006	42.438	0.035
Lasso	-0.085	19860890.263	80.68	0.033
El_Net	-0.062	19689442.759	183.78	0.044
SVR	0.577	12417403.852	278.883	0.775
RFR	0.758558	9791481.605893	787.495	4.375
GBR	0.813	8234405.705	937.707	2.605

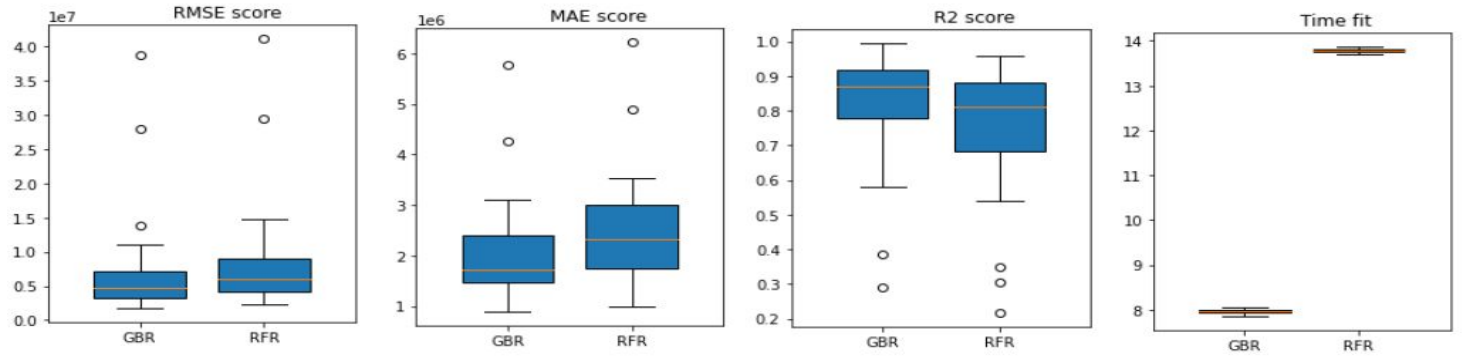
Les modèles linéaires affichent des mauvais scores en géneral sauf le Ridge.

GBR affiche les meilleurs scores sur la grille, le temps d'ajustement moyen (2.6 s) par fit est moins de celui de RFR (4.3s).

Cross-Validation (GBR & RFR)



GHGEmissions

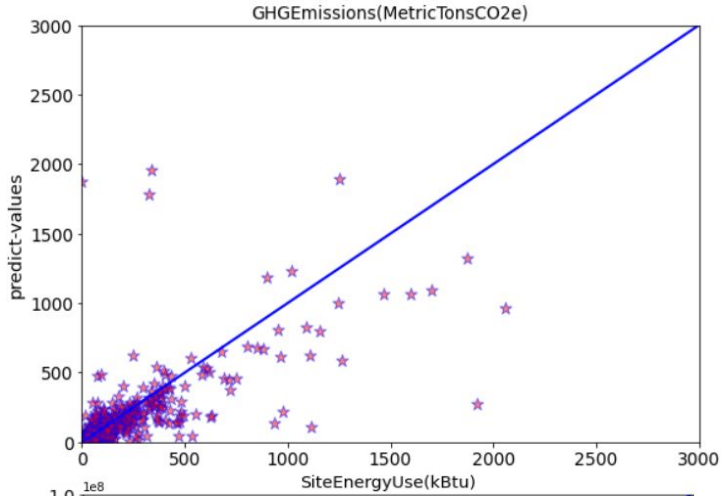


SiteEnergyUse

Résultats du test avec GBR

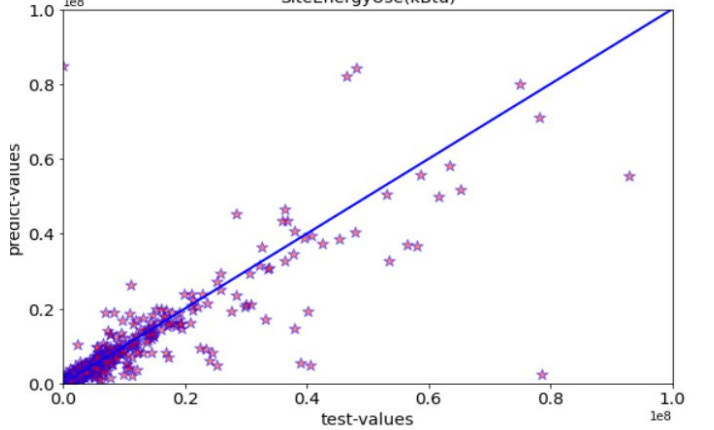
```python
import time
start_time = time.time()
clf_gbr.fit(X_train,y_train['GHGEmissions(MetricTonsCO2e)'])
y_pred = clf_gbr.predict(X_test)
print("time_fit : {:.2} s    ".format((time.time() - start_time)))
model_accuracy(y_pred,y_test['GHGEmissions(MetricTonsCO2e)'])
```

```
time_fit : 8.4 s
MAE (L1)       67.0483
RMSE (L2)      245.3126
Mediane-AE      14.1057
R2 (CV)          0.8084
```



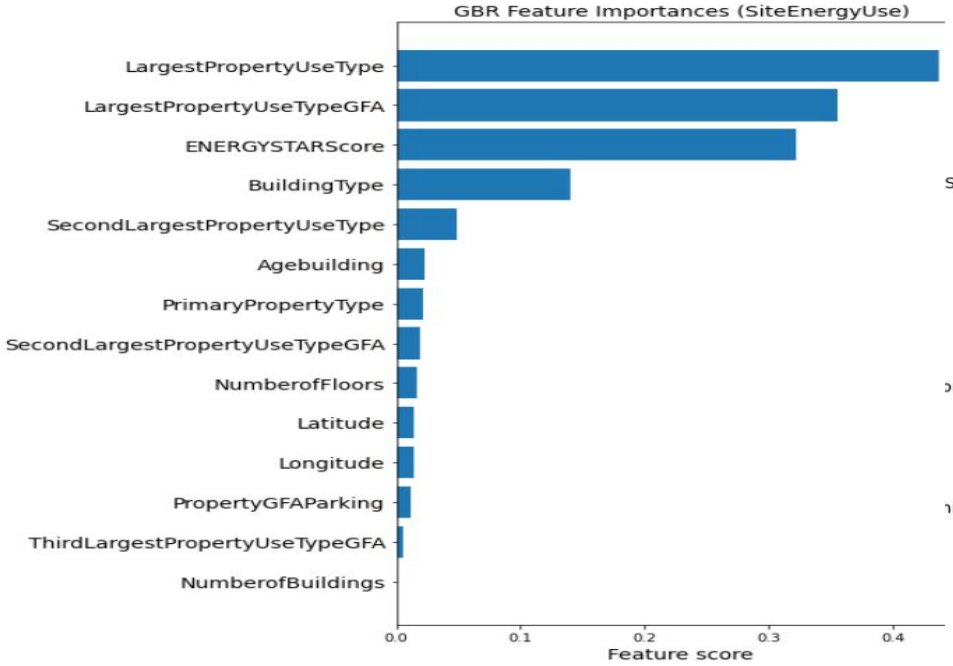

```python
import time
start_time = time.time()
clf_gbr.fit(X_train,y_train['SiteEnergyUse(kBtu)'])
y_pred = clf_gbr.predict(X_test)
print("time_fit : {:.2} s.".format((time.time() - start_time)))
model_accuracy(y_pred,y_test['SiteEnergyUse(kBtu)'])
```

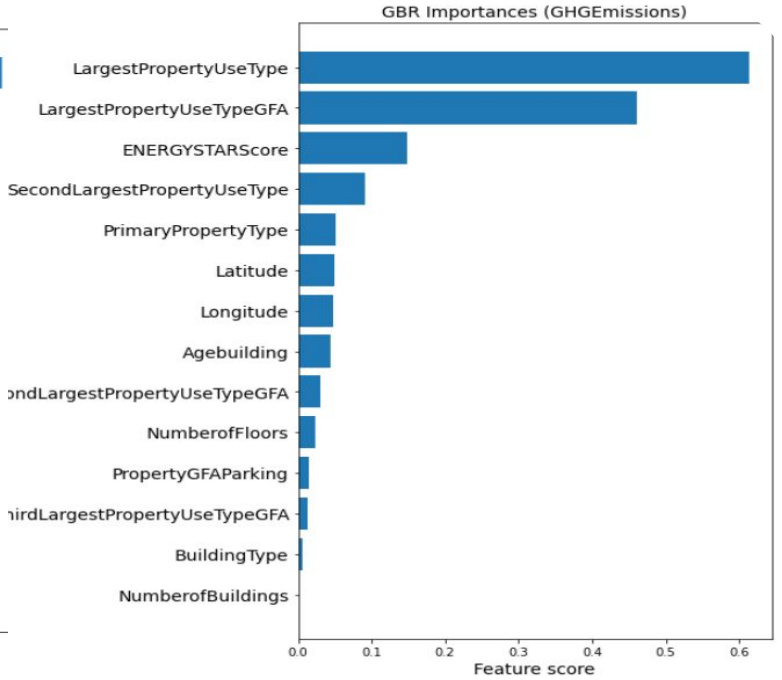
```
time_fit : 1.5e+01 s.
MAE (L1)       2165341.3591
RMSE (L2)      10847709.8982
Mediane-AE      409469.0972
R2 (CV)           0.6763
```



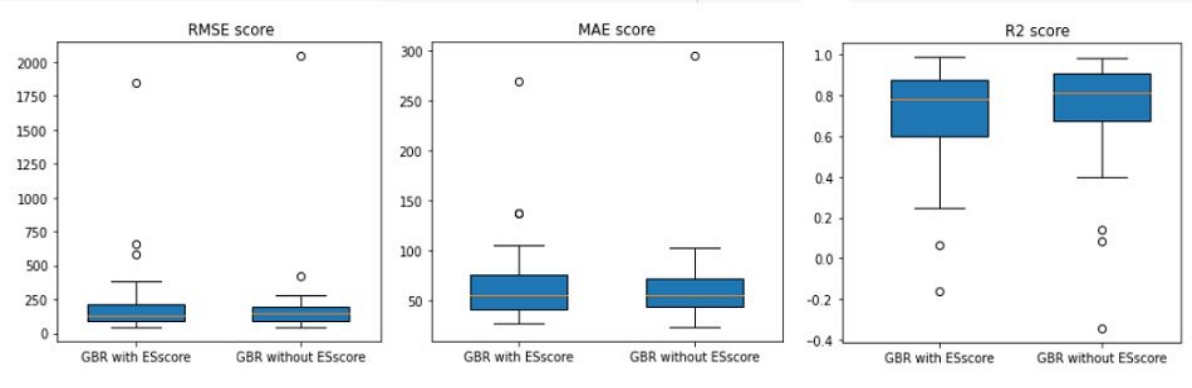
Features Importances

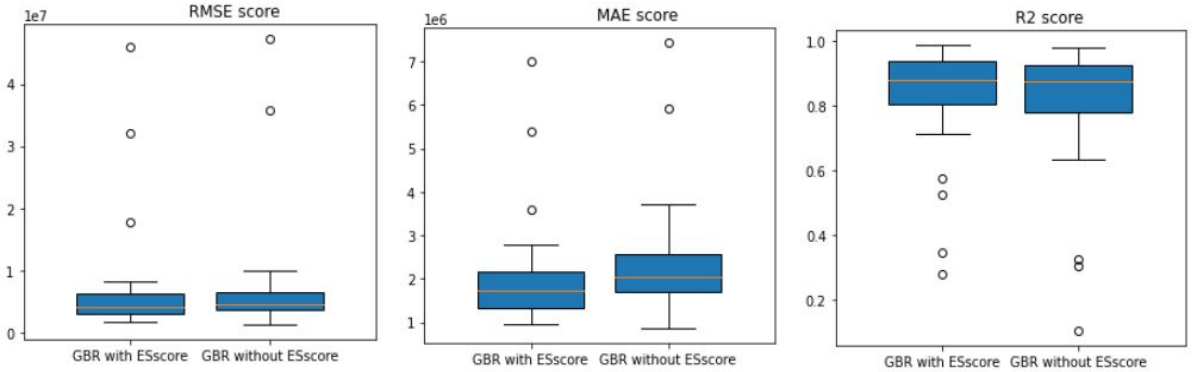
Le type d'utilisation principale (*LargestPropertyUseType*) et la surface large (*LargestPropertyUseTypeGFA*) ont un poids plus important dans les décisions de notre modèle pour les deux variables à prédire. En revanche, le nombre de bâtiments et des étages ont un impact négligeable.





Influence de l'EnergyStarScore





Entraînement du modèle GBR avec et sans ESscore avec Cross-validation

ESscore amélioré très faiblement les scores MAE et R2 pour la cible pour la consommation

Des bons scores sur les données test sans ESscore

Sans intérêt sur les prédictions

Conclusion

GBR est le meilleur régresseur avec

La feature ENERGYSTARScore n'améliore pas de façon significative les prédictions,

amélioration négligeable donc pas de grand intérêt.

Une base de données avec plus d'observations peut être un plus

Autres données techniques peut aider à améliorer les prédictions: (chauffage, isolation, éclairage)

MERCI DE VOTRE ATTENTION