ANTICIPEZ LES BESOINS EN CONSOMMATION DE BÂTIMENTS

BÂTIMENTS NON RÉSIDENTIELS À SEATTLE VERS UNE VILLE NEUTRE EN CARBONE D'ICI 2050

ABBAS Billel

Juin 2024

AGENDA

Seattle

1. Introduction au projet
2. Vue d'ensemble des données & Nettoyage de la données
3. Nettoyage de la données: Filtrage sur les colonnes & sur les lignes
4. Nettoyage de la données: Valeurs aberrantes et extrêmes & valeurs manquantes
5. Feature Engineering simple
6. Feature Engineering complexe
7. TargetEncoder vs OneHotEncoder & Sélection et tester des modèles linéaires
8. Sélection et tester des modèles non linéaires
9. Fine-tuning du meilleur modèle:
10. Analyse de la « feature importance » locale
11. Analyse de la « feature importance » globale
12. Analyse de l'influence de l'EnergyStarScore
13. Conclusion & Amélioration

INTRODUCTION AU PROJET

- **Contexte du projet:**
  - La ville de Seattle s'est fixé comme objectif la neutralité carbone d'ici 2050.
  - Ce projet s'inscrit dans une initiative plus large visant à réduire les émissions de gaz à effet de serre en se concentrant sur les secteurs les plus énergivores, notamment les bâtiments non résidentiels.

- **Problématique:**
  - Face à des données incomplètes sur la consommation d'énergie et les émissions de CO2 de nombreux bâtiments, il est crucial de développer des modèles prédictifs basés sur les données existantes de 2016 pour estimer ces valeurs non observées.

- **Approche:**
  - **Méthodes utilisées :** Application de techniques de machine learning pour analyser les données historiques et prédire les consommations et émissions futures des bâtiments.
  - **Objectif :** Ces modèles permettent de fournir des estimations précises pour informer et orienter les décisions en matière de politiques énergétiques et de rénovations.
  - **Évaluation spécifique** : Analyse de l'utilité de l'"**ENERGY STAR Score**" pour prédire la consommation totale d'énergie et les émissions de CO2.

VUE D'ENSEMBLE DES DONNÉES:

Jeu de données de 2016

Ce jeu de données contient des informations sur :

- ❖ 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)

Taille du dataset :

3376 lignes et 46 variables (21 qualitatives et 25 quantitatives)

Choix de la variable cible:

- La consommation totale d'énergie: SiteEnergyUse(kBtu),
- Les émissions de CO2 : TotalGHGEmissions,

FILTRAGE SUR LES COLONNES:

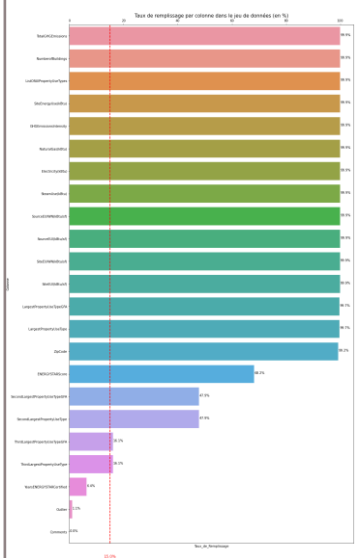
Approche métier:

- **Suppression de variables : 'SiteEnergyUseWN(kBtu)', 'Electricity(kWh)', et 'NaturalGas(therms)'** ont été retirées pour éviter les doublons et les incohérences unitaires, car certaines incluent des ajustements météorologiques ('WN').
- **Filtrage des données:** Seuls les bâtiments non résidentiels ont été conservés, en utilisant les critères **'PrimaryPropertyType' et 'BuildingType',** réduisant ainsi le dataset à 1510 lignes.
- **Élimination de variables non essentielles : 'TaxParcelIdentificationNumber', 'DefaultData**' ont été supprimées pour recentrer l'analyse sur les données plus pertinentes.
- Suppression des variables qui contiennent 'WN': **'SiteEUIWN(kBtu/sf)', 'SourceEUIWN(kBtu/sf)'**

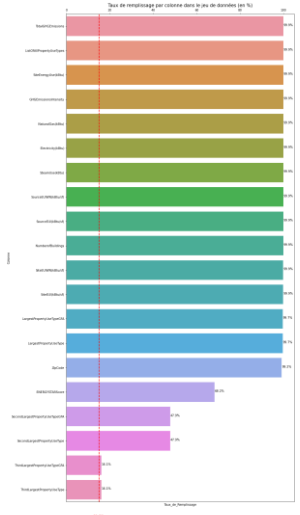
Variables conservées :

- **Qualitatives** (19 au total) : Identifiants et informations de base sur les bâtiments, y compris type, utilisation, localisation, et conformité.
- **Quantitatives** (20 au total) : Mesures de l'emplacement, de la taille, de l'utilisation de l'énergie et des émissions des gaz à effet de serre des propriétés.

Approche technique:



Seuil de 15%



- Après application du taux de seuil de remplissage, les variables **'Outlier', 'YearsENERGYSTARCertified' et 'Comments'** ont été **supprimées.**

FILTRAGE SUR LES LIGNES:

- Suppression des lignes ne présentant pas de valeurs cibles
- Suppression des lignes en doubles

1505 lignes et 36 variables (16 qualitatives et 20 quantitatives)

VALEURS ABERRANTES :

- Suppression des valeurs négatives dans les colonnes 'Electricity(kBtu)', 'GHGEmissionsIntensity' et 'TotalGHGEmissions'.
- Correction effectuée sur la ligne concernant le 'Seattle Chinese Baptist Church' où le nombre d'étages était indiqué comme 99, alors que le plus haut bâtiment de Seattle, le Columbia Center achevé en 1985, en compte seulement 76. Cette valeur a été ajustée à un seul étage.
- Vérification du minimum de la colonne 'NumberofFloors' qui est de 0 étages → car la variable `NumberofFloors` représente le nombre de niveaux où se trouvent ces bâtiments.
- Apres avoir identifier ces valeurs extrêmes, pour mon analyse nous décédons de les conserver.

	NumberofFloors	Electricity(kBtu)	GHGEmissionsIntensity	TotalGHGEmissions
count	1505.000000	1.505000e+03	1505.000000	1505.000000
mean	4.171429	5.808994e+06	1.543674	175.199542
std	6.777808	2.129314e+07	2.170456	657.705265
min	0.000000	-1.154170e+05	-0.020000	-0.800000
25%	1.000000	7.102750e+05	0.340000	19.560000
50%	2.000000	1.602084e+06	0.840000	49.170000
75%	4.000000	4.897760e+06	1.810000	134.570000
max	99.000000	6.570744e+08	25.710000	12307.160000

	NumberofFloors	Electricity(kBtu)	GHGEmissionsIntensity	TotalGHGEmissions
count	1504.000000	1.504000e+03	1504.000000	1504.000000
mean	4.105053	5.812933e+06	1.544714	175.316563
std	6.323472	2.129967e+07	2.170803	657.908353
min	0.000000	0.000000e+00	0.000000	0.000000
25%	1.000000	7.104272e+05	0.340000	19.597500
50%	2.000000	1.603396e+06	0.840000	49.175000
75%	4.000000	4.899381e+06	1.810000	134.600000
max	76.000000	6.570744e+08	25.710000	12307.160000

VALEURS EXTRÊMES:

	Latitude	Longitude	NumberofBuildings	NumberofFloors	PropertyGFATotal	PropertyGFAParking	PropertyGFABuilding(s)	LargestPropertyUseTypeGFA
count	1504.000000	1504.000000	1504.000000	1504.000000	1.504000e+03	1504.000000	1.504000e+03	1.504000e+03
mean	47.615178	-122.333146	1.182846	4.105053	1.209316e+05	12814.392287	1.081172e+05	1.026915e+05
std	0.048931	0.024676	3.079855	6.323472	3.088468e+05	42854.340306	2.958794e+05	2.887343e+05
min	47.499170	-122.411820	0.000000	0.000000	1.128500e+04	0.000000	1.128500e+04	7.583000e+03
25%	47.582900	-122.343428	1.000000	1.000000	2.971400e+04	0.000000	2.871650e+04	2.628025e+04
50%	47.611820	-122.333035	1.000000	2.000000	4.975200e+04	0.000000	4.763750e+04	4.551100e+04
75%	47.649043	-122.322178	1.000000	4.000000	1.062458e+05	0.000000	9.626550e+04	9.404625e+04
max	47.733870	-122.258640	111.000000	76.000000	9.320156e+06	512608.000000	9.320156e+06	9.320156e+06
Q1	47.582900	-122.343428	1.000000	1.000000	2.971400e+04	0.000000	2.871650e+04	2.628025e+04
Q3	47.649043	-122.322178	1.000000	4.000000	1.062458e+05	0.000000	9.626550e+04	9.404625e+04
Borne_superieure	47.748256	-122.290303	1.000000	8.500000	2.210434e+05	0.000000	1.975890e+05	1.956952e+05
Nb_produits_sup_Borne_superieure	0.000000	87.000000	48.000000	127.000000	1.860000e+02	290.000000	1.670000e+02	1.580000e+02

VALEURS MANQUANTES:

Avant traitement:

	Colonne	Taux_de_Remplissage
0	SiteEUI(kBtu/sf)	99.9335
1	ZipCode	99.2021
2	ENERGYSTARScore	68.2846
3	SecondLargestPropertyUseType	48.0718
4	SecondLargestPropertyUseTypeGFA	48.0718
5	ThirdLargestPropertyUseType	16.1569
6	ThirdLargestPropertyUseTypeGFA	16.1569

Approche métier:

Imputation des variables :

- ZipCode : Imputé à partir des données de la colonne Address.
- SecondLargestPropertyUseType et ThirdLargestPropertyUseType: Imputés en utilisant les informations de la colonne ListOfAllPropertyUseTypes.
- SecondLargestPropertyUseTypeGFA et ThirdLargestPropertyUseTypeGFA: Imputés en se basant respectivement sur les colonnes SecondLargestPropertyUseType et ThirdLargestPropertyUseType.

	Colonne	Taux_de_Remplissage
0	SiteEUI(kBtu/sf)	99.9335
1	ENERGYSTARScore	68.2846

Approche technique:

Imputation par iterativeimputer:

Corrélations:

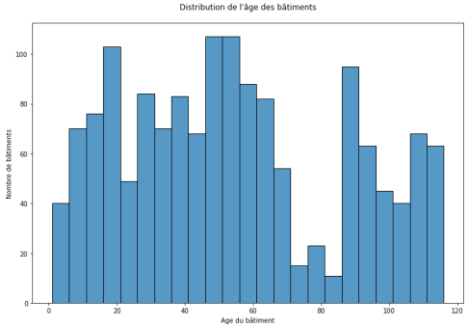
- SiteEUI(kBtu/sf) et SourceEUI(kBtu/sf) 0.94
- SiteEUI(kBtu/sf) et GHGEmissionsIntensity 0.75

	Colonne	Taux_de_Remplissage
0	ENERGYSTARScore	68.2846

Nous n'allons pas imputer la variable restante ENERGYSTARScore car nous devons l'utiliser telle quelle pour évaluer son influence sur la prédiction.

FEATURE ENGINEERING SIMPLE:

- Création de la variable **BuildingAge** (l'âge du bâtiment) à partir de **YearBuilt** (l'année de la construction).
- Création des deux variables la surface moyenne par bâtiment et par étage:
  - GFAPerBuilding = PropertyGFATotal / NumberofBuildings
  - GFAPerFloor = PropertyGFATotal / NumberofFloors
- Calcule de ces proportion:
  - Building(s)_Proportion = PropertyGFABuilding(s) / PropertyGFATotal
  - Parking_Proportion = PropertyGFAParking / PropertyGFATotal
  - LargestPropertyUse_Proportion = LargestPropertyUseTypeGFA / PropertyGFATotal
  - SecondLargestPropertyUse_Proportion = SecondLargestPropertyUseTypeGFA / PropertyGFATotal
  - ThirdLargestPropertyUse_Proportion = ThirdLargestPropertyUseTypeGFA / PropertyGFATotal



Transformation logarithmique (log1) des variables (avec un skewness > 5,5):

- Grâce à cette transformation logarithmique, nous avons pu éviter les erreurs liées aux zéros dans les données.
- Les deux variables 'SiteEnergyUse(kBtu)_log' et 'TotalGHGEmissions_log' sont notre nouvelles variables cibles qui vont remplacer les deux variables 'SiteEnergyUse(kBtu)' et 'TotalGHGEmissions'

'LargestPropertyUseTypeGFA',
'Electricity(kBtu)',
'PropertyGFABuilding(s)',
'SiteEnergyUse(kBtu)',
'PropertyGFATotal',
'SteamUse(kBtu)',
'ThirdLargestPropertyUseTypeGFA',
'NaturalGas(kBtu)',
'TotalGHGEmissions',
'SecondLargestPropertyUseTypeGFA'

Transformation en log

'LargestPropertyUseTypeGFA_log',
'Electricity(kBtu)_log',
'PropertyGFABuilding(s)_log',
'SiteEnergyUse(kBtu)_log',
'PropertyGFATotal_log',
'SteamUse(kBtu)_log',
'ThirdLargestPropertyUseTypeGFA_log',
'NaturalGas(kBtu)_log',
'TotalGHGEmissions_log',
'SecondLargestPropertyUseTypeGFA_log'

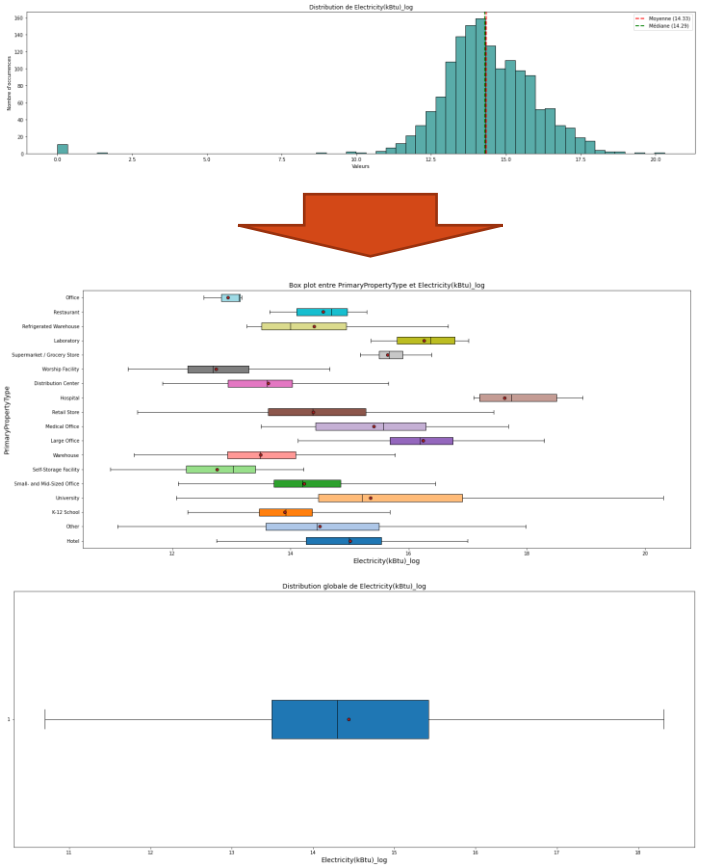
Vérification du data leakage avec les variables cibles:

- **SiteEnergyUse(kBtu)_log et TotalGHGEmissions_log:** suppression des variables ['PropertyGFATotal_log','PropertyGFABuilding(s)_log', 'LargestPropertyUseTypeGFA_log', 'SecondLargestPropertyUseTypeGFA_log', 'ThirdLargestPropertyUseTypeGFA_log', 'SiteEUI(kBtu/sf)', 'SourceEUI(kBtu/sf)', 'GHGEmissionsIntensity'].

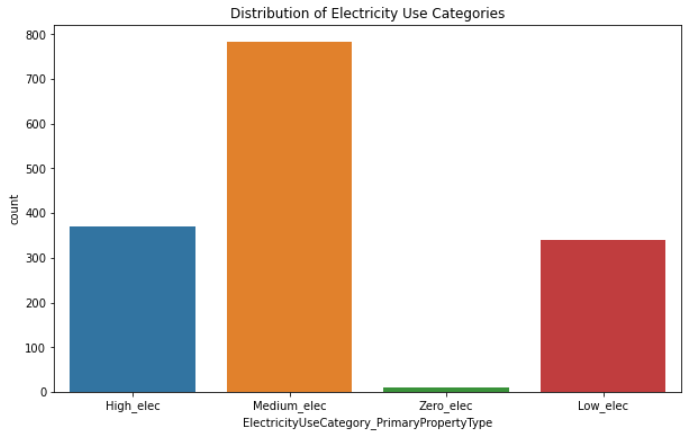
FEATURE ENGINEERING COMPLEXE:

- **Création de nouvelles variables tenant compte de :**
  - L'effet du type de propriété sur la consommation d'énergie.
  - L'effet du type de bâtiment sur la consommation d'énergie.
  - L'effet de l'âge du bâtiment sur la consommation d'énergie.
  - L'effet du voisinage sur la consommation d'énergie.

- **On prend un exemple de l'effet PrimaryPropertyType sur Electricity(kBtu)_log:**



- **Zero_elec Category:** quand la valeurs de la variable Electricity(kBtu)_log = 0
- **High_elec Category:** Hotel, University, Large Office, Medical Office, Hospital, Supermarket / Grocery Store, Laboratory.
- **Medium_elec Category:** Other, K-12 School, Small- and Mid-Sized Office, Retail Store, Refrigerated Warehouse, Restaurant.
- **Low_elec Category:** Self-Storage Facility, Warehouse, Distribution Center, Worship Facility, Office.



- **Le nombre de variables qui ont été supprimé après le future engineering est: 45**
- **Le nombre de variables qui ont été crée grâce aux future engineering est: 20**

TARGETENCODER VS ONEHOTENCODER:

```python
# Pipeline pour les variables numériques
numeric_transformer = Pipeline(steps=[
    ('imputer', SimpleImputer(strategy='mean')),
    ('scaler', StandardScaler())
])

# Préprocesseurs et pipelines pour Target et One Hot Encoding
preprocessors = {
    'target': ColumnTransformer(transformers=[
        ('num', numeric_transformer, numeric_features),
        ('cat', Pipeline(steps=[
            ('imputer', SimpleImputer(strategy='most_frequent')),
            ('target_encoder', TargetEncoder())
        ]), categorical_features)
    ]),
    'onehot': ColumnTransformer(transformers=[
        ('num', numeric_transformer, numeric_features),
        ('cat', Pipeline(steps=[
            ('imputer', SimpleImputer(strategy='most_frequent')),
            ('onehot', OneHotEncoder(handle_unknown='ignore'))
        ]), categorical_features)
    ])
}
```

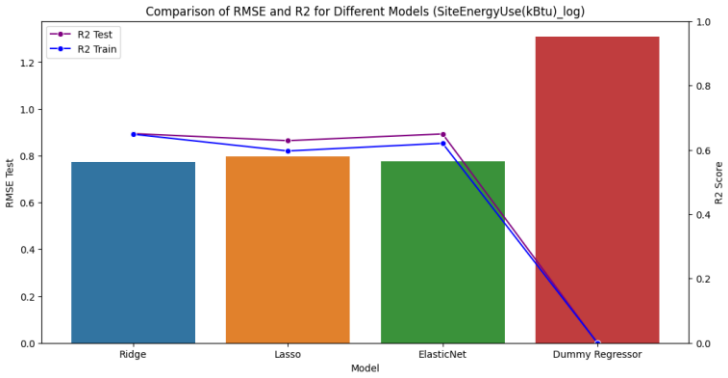
- **Results for RandomForest:** {**'target':** {'R2': 0.746, 'RMSE': 0.633}, **'onehot':** {'R2': 0.755, 'RMSE': 0.621}}
- **Results for XGBoost:** {**'target':** {'R2': 0.724, 'RMSE': 0.660}, **'onehot':** {'R2': 0.740, 'RMSE': 0.640}}
- **Results for GradientBoost:** {**'target':** {'R2': 0.737, 'RMSE': 0.644}, **'onehot':** {'R2': 0.763, 'RMSE': 0.611}}
- **One Hot Encoding** est la meilleure méthode selon le R^2 et le RMSE.

SÉLECTION ET TESTER DES MODÈLES LINÉAIRES

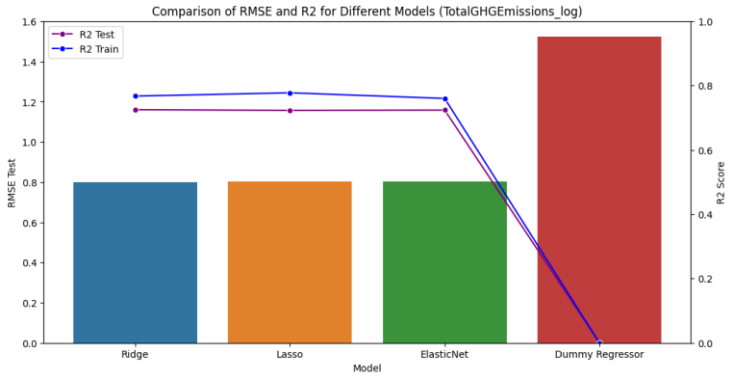
Séparation en variables catégorielles et numériques:

- Variables catégorielles: **imputer → most_frequent , OneHotEncoder**.
- Variables numériques: **imputer → mean, StandardScaler.**

Pour la consommation d'énergie



Pour l'émission Co2



Le modèle Ridge est le meilleure en terme de RMSE teste et R2 test.

SÉLECTION ET TESTER DES MODÈLES NON LINÉAIRES

Séparation en variables catégorielles et numériques:

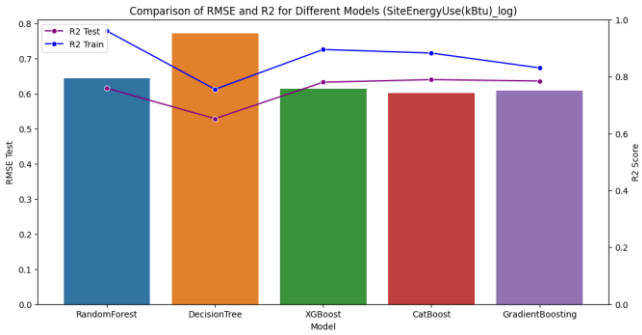
Variables catégorielles: **imputer → most_frequent , OneHotEncoder**.

Variables numériques: **imputer → mean, StandardScaler.**

```
models = {
    'RandomForest': (RandomForestRegressor(), {
        'regressor__n_estimators': [200, 250, 300],
        'regressor__max_depth': [30, 35, 40]
    }),
    'DecisionTree': (DecisionTreeRegressor(), {
        'regressor__max_depth': [10, 15, 20],
        'regressor__min_samples_split': [20, 25, 30],
        'regressor__min_samples_leaf': [5, 7, 10]
    }),
    'XGBoost': (XGBRegressor(), {
        'regressor__n_estimators': [100, 150, 200],
        'regressor__learning_rate': [0.1, 0.15, 0.2],
        'regressor__max_depth': [3, 4, 5]
    }),
    'CatBoost': (CatBoostRegressor(silent=True), {
        'regressor__iterations': [300, 350, 400],
        'regressor__learning_rate': [0.05, 0.1, 0.15],
        'regressor__depth': [4, 5, 6]
    }),
    'GradientBoosting': (GradientBoostingRegressor(), {
        'regressor__n_estimators': [100, 150, 200],
        'regressor__learning_rate': [0.05, 0.1, 0.15],
        'regressor__max_depth': [3, 4, 5]
    })
}
```

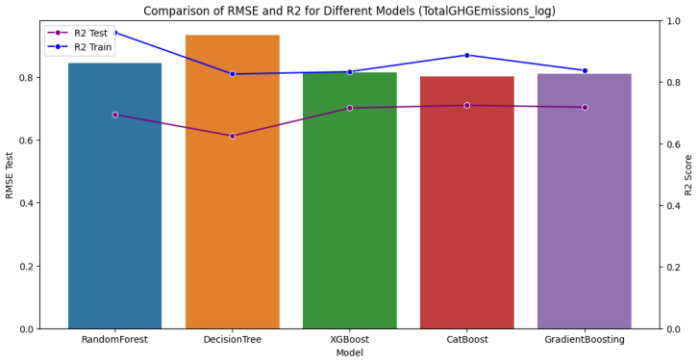
Liste des modèles avec leurs paramètres pour **GridSearchCV**

Pour la consommation d'énergie



Best model for SiteEnergyUse(kBtu)_log: CatBoost with:
RMSE Test: 0.601, R2 Test: 0.789, R2 Train: 0.882

Pour l'émission Co2



Best model for TotalGHGEmissions_log: CatBoost with:
RMSE Test: 0.801, R2 Test: 0.723, R2 Train: 0.887

FINE-TUNING DU MEILLEUR MODÈLE:

```python
# Mise à jour des hyperparamètres pour un fine-tuning plus ciblé
models = {
    'CatBoost': (CatBoostRegressor(random_state=42, verbose=0), {
        'iterations': np.arange(480, 520, 5),
        'learning_rate': np.linspace(0.045, 0.055, 11),
        'depth': [5, 6, 7],
        'l2_leaf_reg': np.linspace(0.5, 3.0, 6),
        'bagging_temperature': np.linspace(0.1, 0.4, 4),
        'border_count': [50, 100, 150, 200],
        'boosting_type': ['Plain'],
        'rsm': [0.8, 0.85, 0.9, 0.95, 1.0]
    })
}
```

Utiliser **RandomizedSearchCV** pour l'ajustement fin, une méthode plus rapide en temps de calcule pour trouver les meilleurs hyperparamètres du modèle **CatBoost**.

```python
# Effectuer le fine-tuning pour chaque modèle
for model_name, (model, param_dist) in models.items():
    print(f"Fine-tuning pour le modèle: {model_name}")

    # Utiliser RandomizedSearchCV pour le fine-tuning
    random_search = RandomizedSearchCV(
        estimator=model,
        param_distributions=param_dist,
        n_iter=50,  # Nombre d'itérations à effectuer
        scoring='neg_mean_squared_error',
        cv=5,
        verbose=1,
        random_state=42,
        n_jobs=-1
    )

    # Ajuster le modèle
    random_search.fit(X_train_transformed, y_train)
    params = random_search.best_params_

    # Meilleurs hyperparamètres
    print(f"Meilleurs hyperparamètres pour {model_name}: {random_search.best_params_}")
```

Pour la consommation d'énergie

Results for SiteEnergyUse(kBtu)_log:

Before fine_tuning:
CatBoost - RMSE Test: 0.6013, R2 Test: 0.7890, R2 Train: 0.8822

After fine_tuning:
CatBoost - RMSE Test: 0.5888, R2 Test: 0.7977, R2 Train: 0.9159,

Best Params: {
'regressor__bagging_temperature': 0.2,
'regressor__boosting_type': 'Plain',
'regressor__border_count': 150,
'regressor__depth': 5,
'regressor__iterations': 495,
'regressor__l2_leaf_reg': 0.5,
'regressor__learning_rate': 0.054,
'regressor__rsm': 0.8}

Pour l'émission Co2

Results for TotalGHGEmissions_log:

Before fine_tuning:
CatBoost - RMSE Test: 0.801, R2 Test: 0.723, R2 Train: 0.887

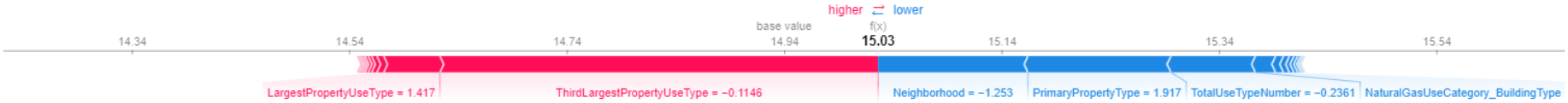
After fine_tuning:
CatBoost - RMSE Test: 0.7856, R2 Test: 0.7344, R2 Train: 0.8288,

Best Params: {
'regressor__bagging_temperature': 0.2,
'regressor__boosting_type': ' Ordered ',
'regressor__border_count': 150,
'regressor__depth': 5,
'regressor__iterations': 400,
'regressor__l2_leaf_reg': 4.0,
'regressor__learning_rate': 0.05,
'regressor__rsm': 0.9}

ANALYSE DE LA « FEATURE IMPORTANCE » LOCALE

Pour la consommation d'énergie

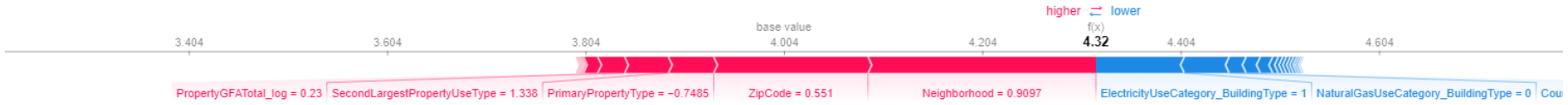
Row = 3



- **Le modèle CatBoost** a prédit une consommation d'énergie de **15.03.**
- **Variables contribuent a augmenter la prédiction:** LargestPropertyUseType, ThirdLargestPropertyUseType
- **Variables contribuent a diminuer la prédiction:** Neighborhood, PrimaryPropertyType, TotalUseTypeNumber, NaturalGasUseCategory_BuildingType

Pour l'émission Co2

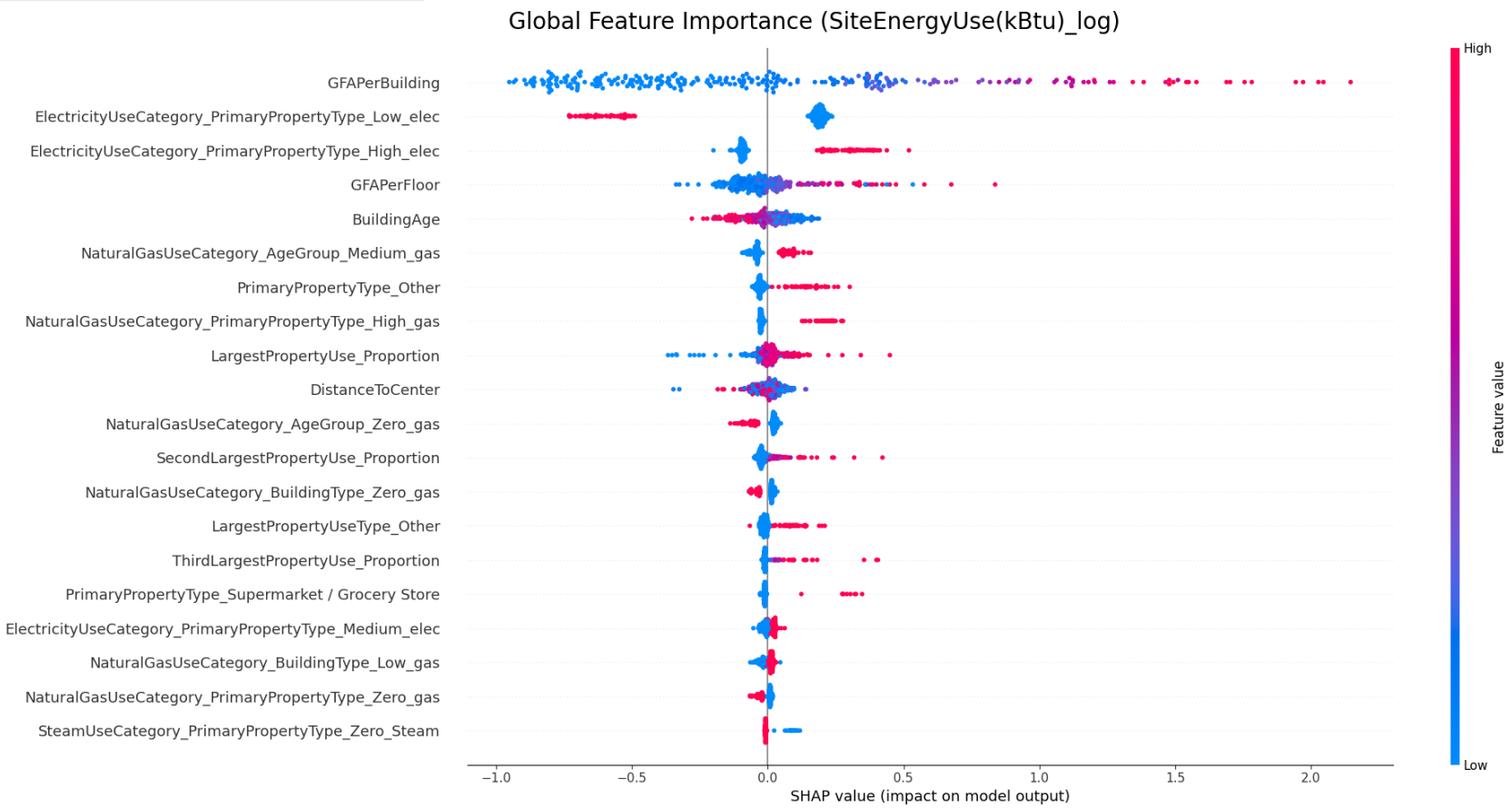
Row = 3



- **Le modèle CatBoost** a prédit une émission totale de gaz à effet de serre de **4.32.**
- **Variables contribuent a augmenter la prédiction:** PropertyGFATotal_log, Neighborhood, ZipCode, PrimaryPropertyType, SecondLargestPropertyUseType
- **Variables contribuent a diminuer la prédiction:** ElectricityUseCategory_BuildingType, NaturalGasUseCategory_BuildingType

ANALYSE DE LA « FEATURE IMPORTANCE » GLOBALE

Pour la consommation d'énergie

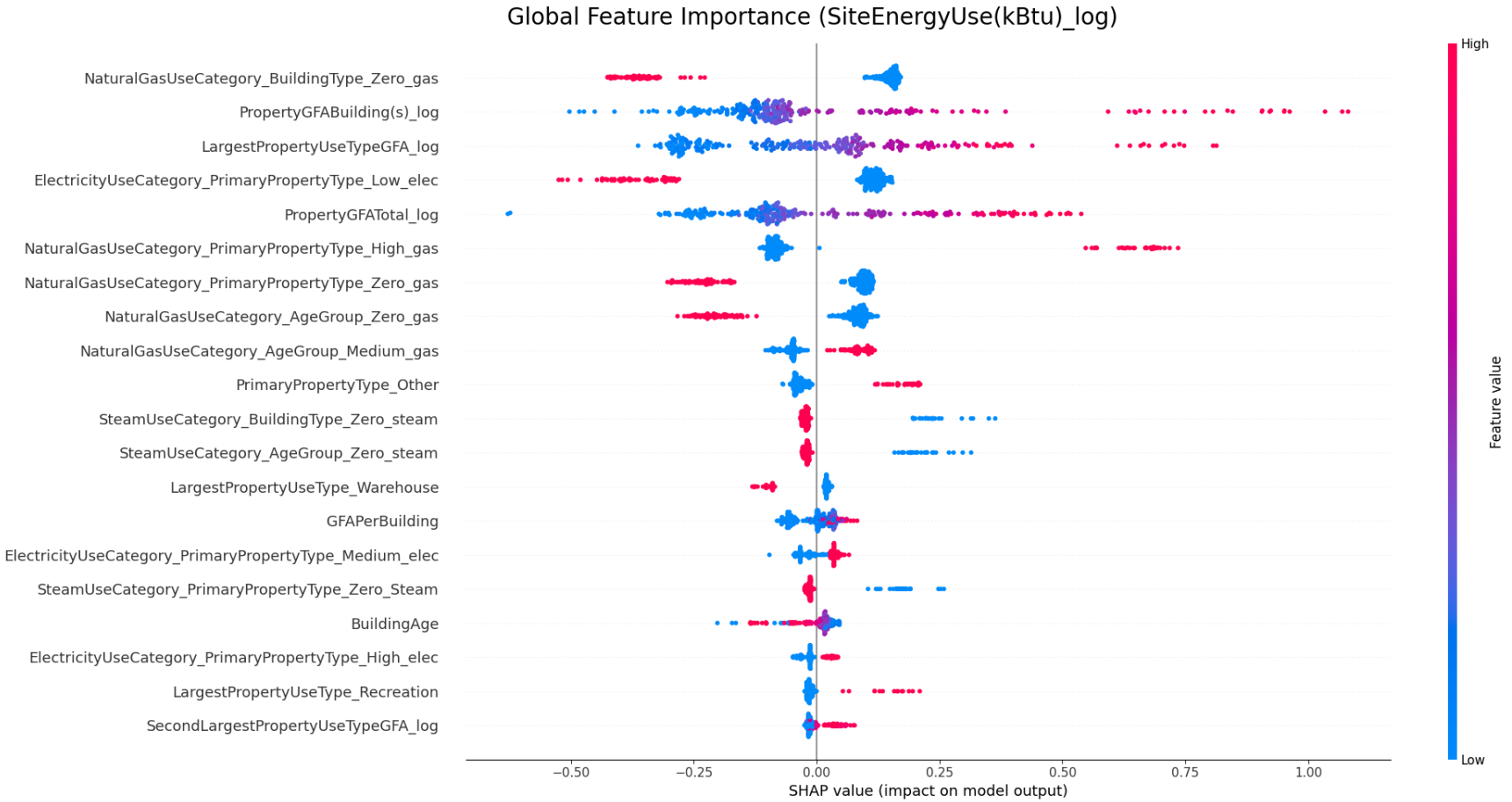


Caractéristiques Principales :

- **GFA Per Building (Surface de Plancher Par Bâtiment):** un impact Très influent
- **Electricity Use Category for Primary Property Type - High and Low Electricity Use:** un impact qui influencent significativement la prédiction.
- **GFA Per Floor (Surface de Plancher Par Étage):** un impact qui exerce une grande influence positive.
- **Natural Gas Use Category for Primary Property Type - High Gas Use:** un Impact qui est fortement positif.

ANALYSE DE LA « FEATURE IMPORTANCE » GLOBALE

Pour l'émission Co2



Caractéristiques Principales :

- **NaturalGasUseCategory_BuildingType_Zero_gas:** un impact Très influent
- **PropertyGFABuilding(s)_log (Surface de Plancher par Bâtiment):** un impact qui influencent significativement la prédiction.
- **LargestPropertyUseTypeGFA_log:** un impact Considérable.
- **ElectricityUseCategory_PrimaryPropertyType_Low_elec:** un Impact Notable.

ANALYSE DE L'INFLUENCE DE L'ENERGYSTARSCORE

Pour la consommation d'énergie

Sans EnergyStarScore:
CatBoost - RMSE Test: 0.533, R2 Test: 0.824, R2 Train: 0.958

Avec EnergyStarScore:
CatBoost - RMSE Test: 0.533, R2 Test: 0.824, R2 Train: 0.958

L'intégration de l'EnergyStarScore dans le modèle CatBoost n'a pas modifié les performances pour **prédire** la consommation d'énergie **(SiteEnergyUse(kBtu)).**

Pour l'émission Co2

Sans EnergyStarScore:
CatBoost - RMSE Test: 0.631, R2 Test: 0.785, R2 Train: 0.868

Avec EnergyStarScore:
CatBoost - RMSE Test: 0.631, R2 Test: 0.785, R2 Train: 0.868

L'ajout de l'EnergyStarScore n'a pas modifié les performances du modèle CatBoost pour **la prédiction** des émissions totales de gaz à effet de serre **(TotalGHGEmissions_log).**

Ces résultats indiquent que **l'EnergyStarScore n'a pas eu d'impact significatif** sur la précision ou la généralisation du modèle pour **la consommation d'énergie ou l'émission de CO2.**

CONCLUSION

- Les informations dont l'on dispose nous permettent d'avoir une prédiction de la consommation d'énergie.
- Le CatBoostRegressor est notre algorithme le plus performant.
- On a cherché à optimiser les paramètres de ces différents algorithmes par le biais d'une validation croisée
- L'ajout d'une variable supplémentaire comme le score Energy Star a modifié les scores de la prédiction.

AMÉLIORATION

- 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.