Projet 3 : Anticipez les besoins en consommation de bâtiments

But : prédiction des émissions de CO2 et consommation totale d'énergie des bâtiments non destinés à l'habitation, évaluation de l'intérêt de l'ENERGY STAR Score pour la prédiction

Compétences : apprentissage supervisé, transformation de variables, validation de performances

Tables des matières

I. Analyse exploratoire et préparation des données

a. Gestion des valeurs manquantes et aberrantes

traitement des **valeurs manquantes :**

- **suppression** des individus comprenant trop de valeurs manquantes (seuil 90%)
- **imputation par le mode ou la médiane**
- imputation par 0 si surface non renseignée
- classe ‘None’ pour les types d’habitation non renseignées (hypothèse de non-existence)

traitement des **valeurs aberrantes :**

- correction des valeurs nulles d’après connaissance métiers

I. Analyse exploratoire et préparation des données
b. Gestion des valeurs atypiques et premier filtrage

Traitement des **valeurs atypiques** :

- suppression des *Outliers* high et low
- suppression des individus dont les valeurs cibles sont atypiques en considèrent l'interquartile :

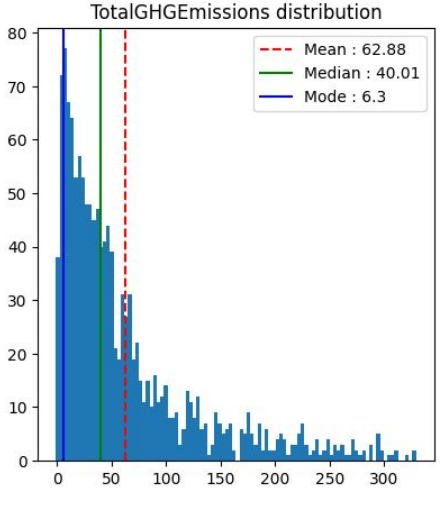
$$\text{Lower} = Q1 - 1.5 * IQR$$

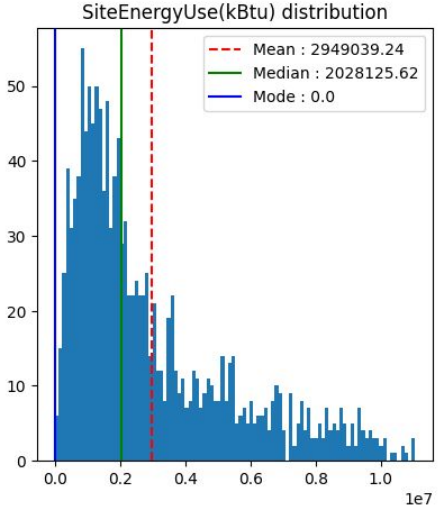
$$\text{Upper} = Q3 + 1.5 * IQR$$

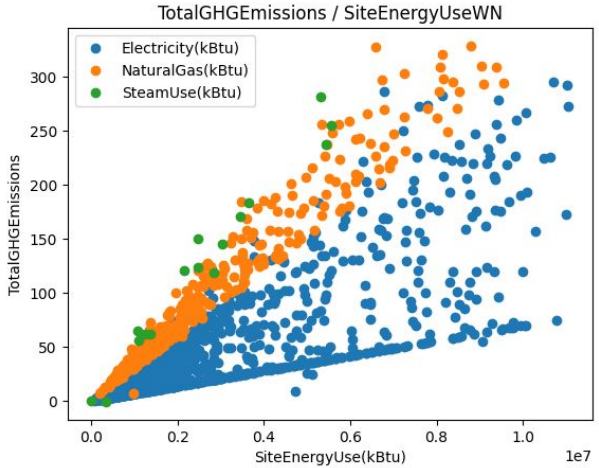
Premiers filtrages :

- Garder uniquement les *ComplianceStatus* sans erreur et sans valeurs manquantes
- Garder uniquement les bâtiments non destinés à l'habitation.
- suppression des variables a une seule modalité et donc n'apportant pas d'information : *City*, *State*, *DataYear*
- suppression des variables qui présentent peu d'intérêt pour notre étude : *PropertyName, TaxParcelIdentificationNumbe, DefaultData*

I. Analyse exploratoire et préparation des données
c. Sélection des variables cibles







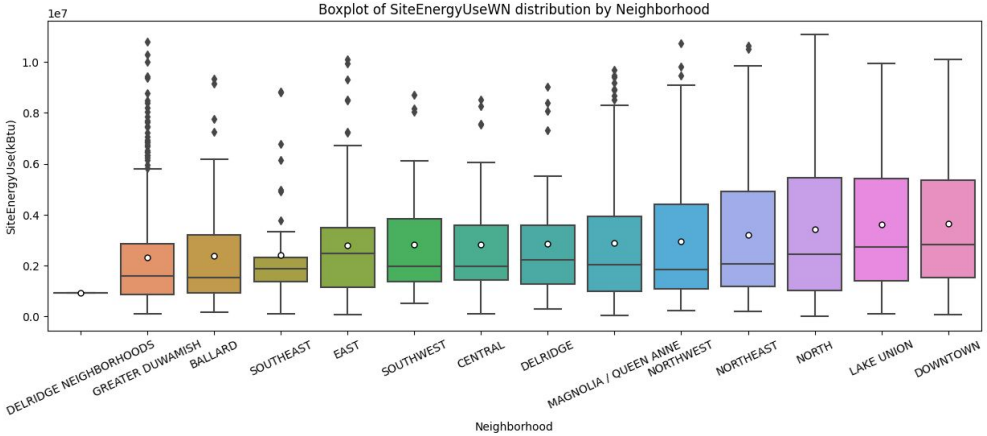
variables corrélées, pearson = 0.74

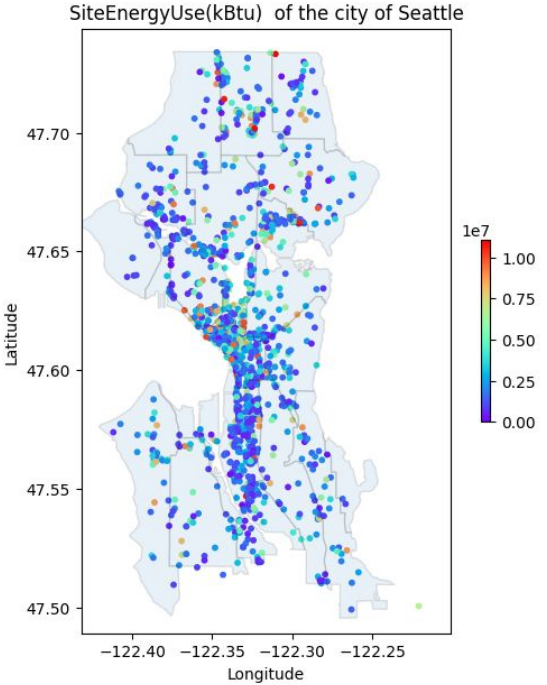
```
eData.loc[eData['SteamUse(kBtu)']     > 0 , 'SteamUsed']          = True
eData.loc[eData['Electricity(kBtu)'] > 0 , 'ElectricityUsed']   = True
eData.loc[eData['NaturalGas(kBtu)' ] > 0 , 'NaturalGasUsed']    = True
```

I. Analyse exploratoire et préparation des données

a. Analyse de la localisation et du type de bâtiment

Prise en compte de la localisation du bâtiment ?





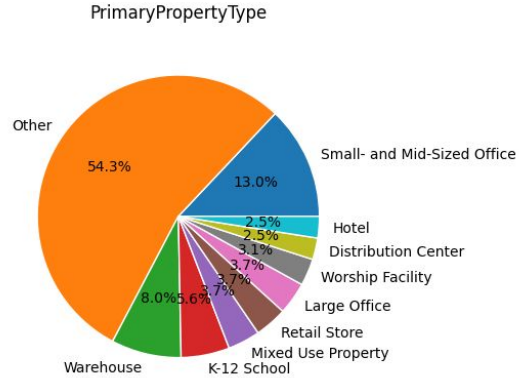
Levene test -> hypothèse homoscédasticité pas satisfaite, on ne peut pas poursuivre avec ANOVA

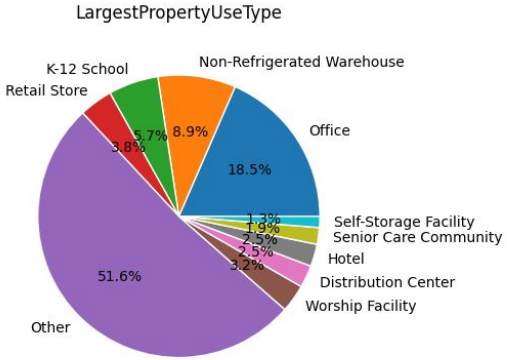
Kruskal-Wallis test -> consommation d'énergie est différentes selon les quartiers donc intérêt de sa prise en compte

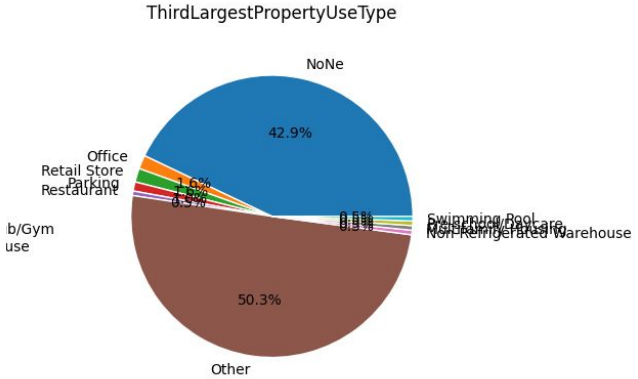
I. Analyse exploratoire et préparation des données

a. Analyse de la localisation et du type de bâtiment

Prise en compte de la localisation du bâtiment ?







	Primary GFA (%)	Second GFA (%)	Third GFA (%)
Hospital (General Medical & Surgical)	200.0	0.0	0.0
Parking	65.79	33.33	6.7
Other - Utility	94.15	5.14	0.0
Senior Care Community	83.16	0.0	0.0
K-12 School	103.39	0.0	0.0
Hotel	88.88	0.0	0.0
Residence Hall/Dormitory	81.57	0.0	0.0
Strip Mall	96.75	0.0	0.0
College/University	102.28	0.0	0.0
Office	76.79	14.3	0.0
Distribution Center	90.18	0.0	0.0
Non-Refrigerated Warehouse	83.66	0.0	0.0
Retail Store	88.32	0.0	0.0
Laboratory	95.07	30.27	0.0
Medical Office	74.14	0.0	0.0

$$PrimaryGFA(\%) = \frac{Median of LargestPropertyUseTypeGFA}{Median of PropertyGFATotal} * 100$$

$$SecondGFA(\%) = \frac{Median of SecondLargestPropertyUseTypeGFA}{Median of PropertyGFATotal} * 100$$

$$ThirdGFA(\%) = \frac{Median of ThirdLargestPropertyUseTypeGFA}{Median of PropertyGFATotal} * 100$$

II. Transformation de variables :

a. Processus itératif

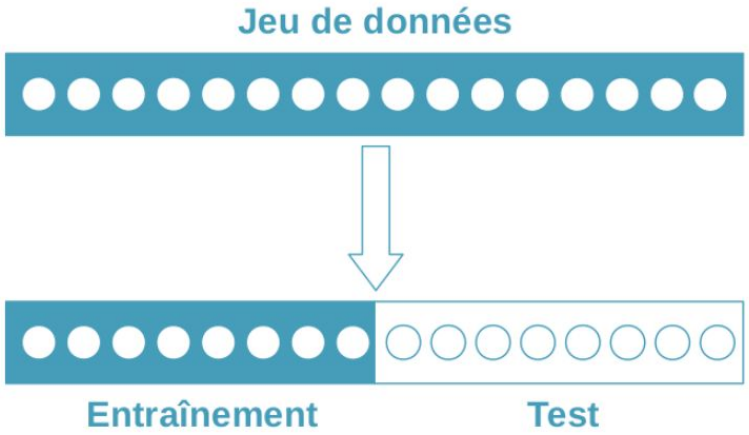
transformation variable : **processus itératif**

- transformation en variables numériques
- la **normalisation**
- **se rapprocher distribution normale**

“memory leackage” ou fuite de données

On sépare donc les données en deux jeux distincts grâce à la fonction **train_test_split** de **sklearn.model_selection : 70 % - 30 %**

L’intérêt des transformations est évalué puis re-effectué si besoin en fonction du score obtenu pour la régression linéaire.



II. Transformation de variables :

b. Gestion des variables catégorielles et nouvelles variables

Variables catégorielles de type :

- **chaine de caractères** :
- **booléen**

testé :

- **OneHotEncoding**
- **Target Encoding** → choix par processus itératif

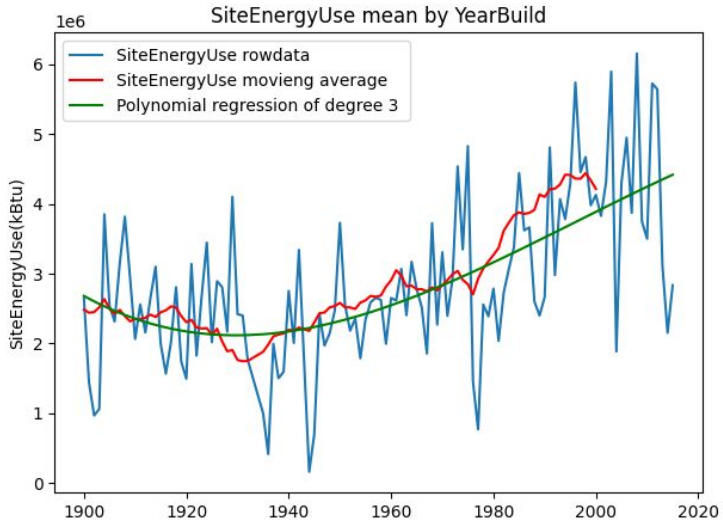
```
X_train[new_name] = X_train[col].map(target_encoding)
X_test[new_name] = X_test[col].map(target_encoding)
# Handle residual : some testing set modality might not have been encountered is the training set
# therefore we will take the median value for these cases
X_test.loc[X_test[new_name].isna(), new_name]  = train.mean()[target]
```

II. Transformation de variables :

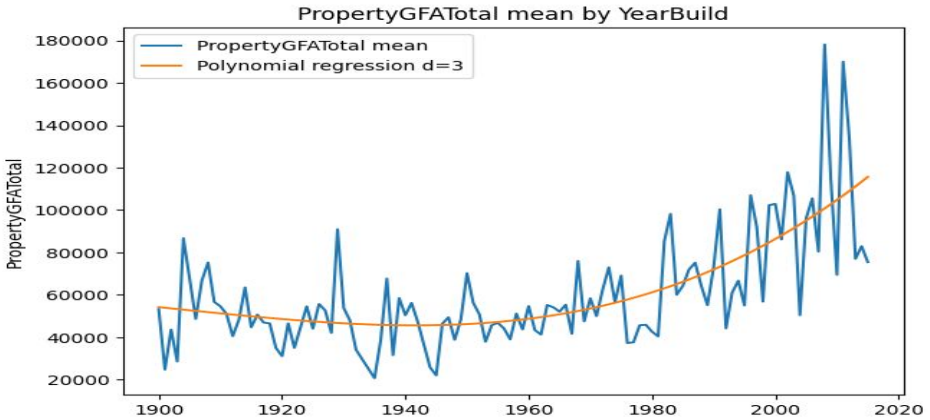
b. Gestion des variables catégorielles et nouvelles variables

$$LargestPropertyUseType\_byGFA = \frac{LargestPropertyUseType\_encoded * LargestPropertyUseTypeGFA}{PropertyGFATotal}$$

de même pour ***SecondLargestPropertyUseType_byGFA*** et *ThirdLargestPropertyUseType_byGFA*



→ nouvelle variable ***YearBuilt_trend***



II. Transformation de variables :

c. Transformation des variables

Gestion de l'asymétrie :

Si l'asymétrie est supérieure à un certain seuil évalué graphiquement, on procède à une transformation Box Cox spécifique à chaque variable et calculé sur le jeu d'entrainement.

$$B(x,\lambda) = \begin{cases} \frac{x^{\lambda}-1}{\lambda} & \text{si } \lambda \neq 0 \\ \log(x) & \text{si } \lambda = 0 \end{cases}$$

Standardisation pour variables en entrée et les cibles :
MinMaxScaler(), données entre 0 et 1 (Robust et Standard ont également été testés)

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

II. Prédiction

a. Choix des modèles et adaptation des hyper-paramètres par validation croisées

➔ modèle dit “naif” : **dummy** regresseur de sklearn

➔ Régression linéaire multiple

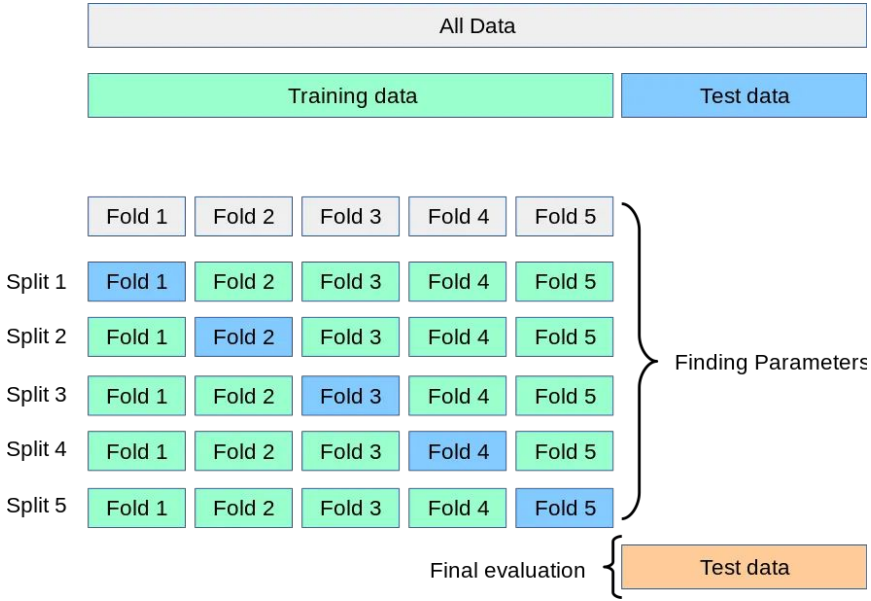
➔ Régression régularisée : Lasso

➔ Modèle non-linéaire : SVR a noyau rbf (Noyau de fonction de base radiale)

➔ Modèle ensembliste : apprenants faibles : forets aléatoires **RadomForestRegressor**

II. Prédiction

b. Adaptation des hyper-paramètres par validation croisée



- Repeated KFold
- GridSearch
- Cross-validation

choix des hyperparametres en maximisant R2 :

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

III. Prédiction

b. Adaptation des hyper-paramètres par validation croisée

➔ Régression régularisée : Lasso
- **gamma ou lambda** : coefficient de régularisation

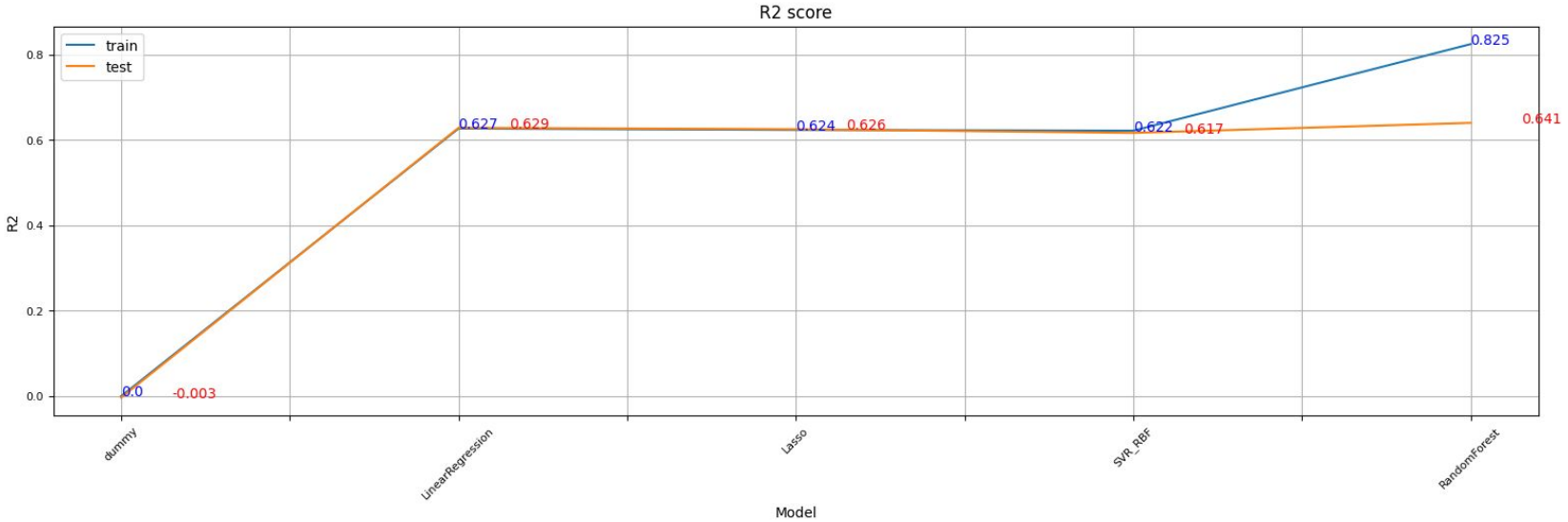
➔ Modèle non-linéaire : SVR a noyau rbf
- **C** : paramètre de pénalité
- **gamma :** kernel parameter

➔ Modèle ensembliste : Random Forest
- **taille** : nombre d'arbres
- **profondeur des arbres** : nombre de branches par arbre

III. Prédiction

c. Évaluation, choix du modèle final et interprétable

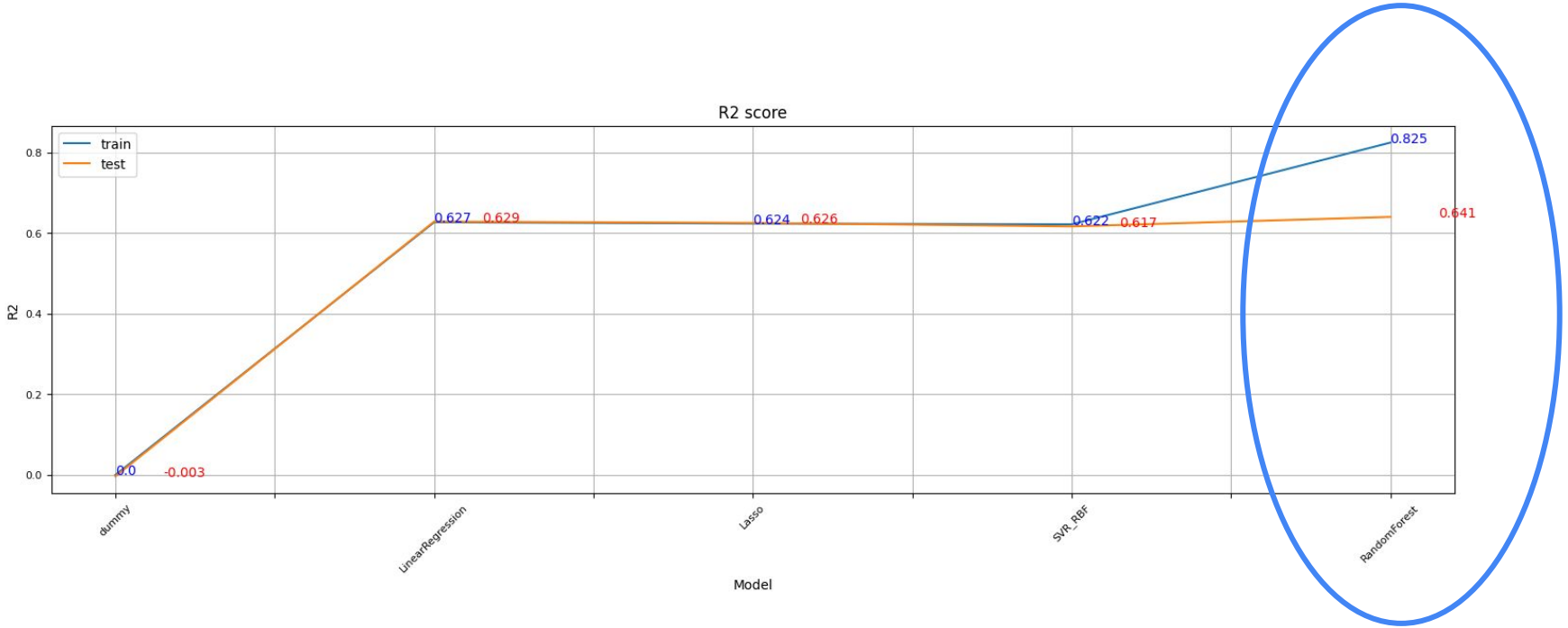
Prédiction de la Consommation d'Énergie



III. Prédiction

c. Évaluation, choix du modèle final et interprétable

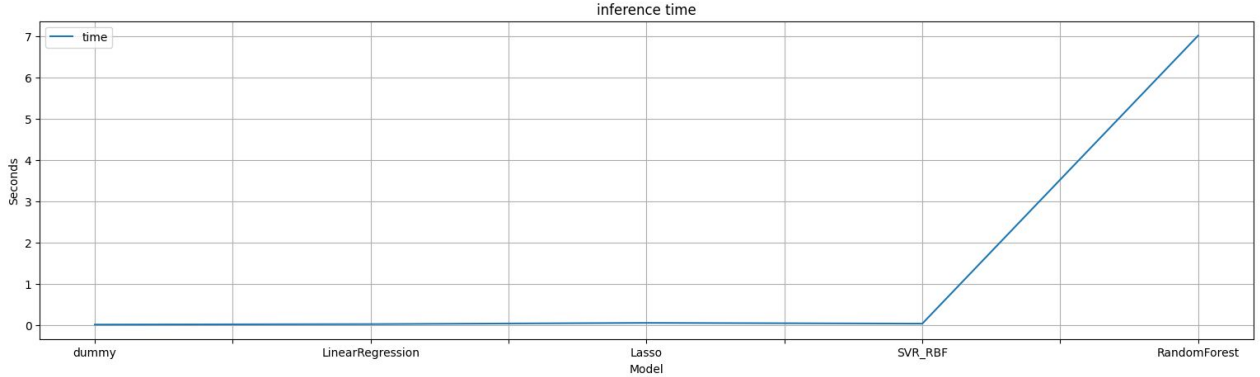
Prédiction de la Consommation d'Énergie

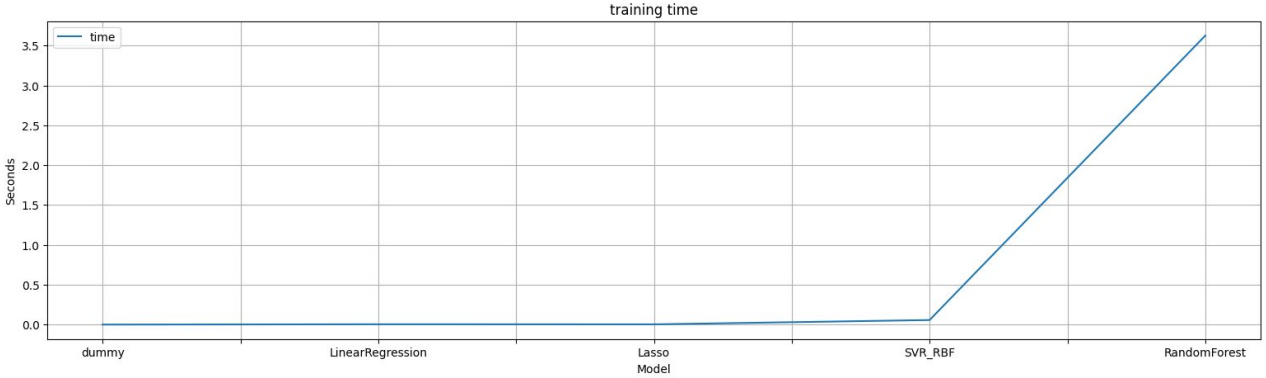


III. Prédiction

c. Évaluation, choix du modèle final et interprétable

Prédiction de la Consommation d'Énergie



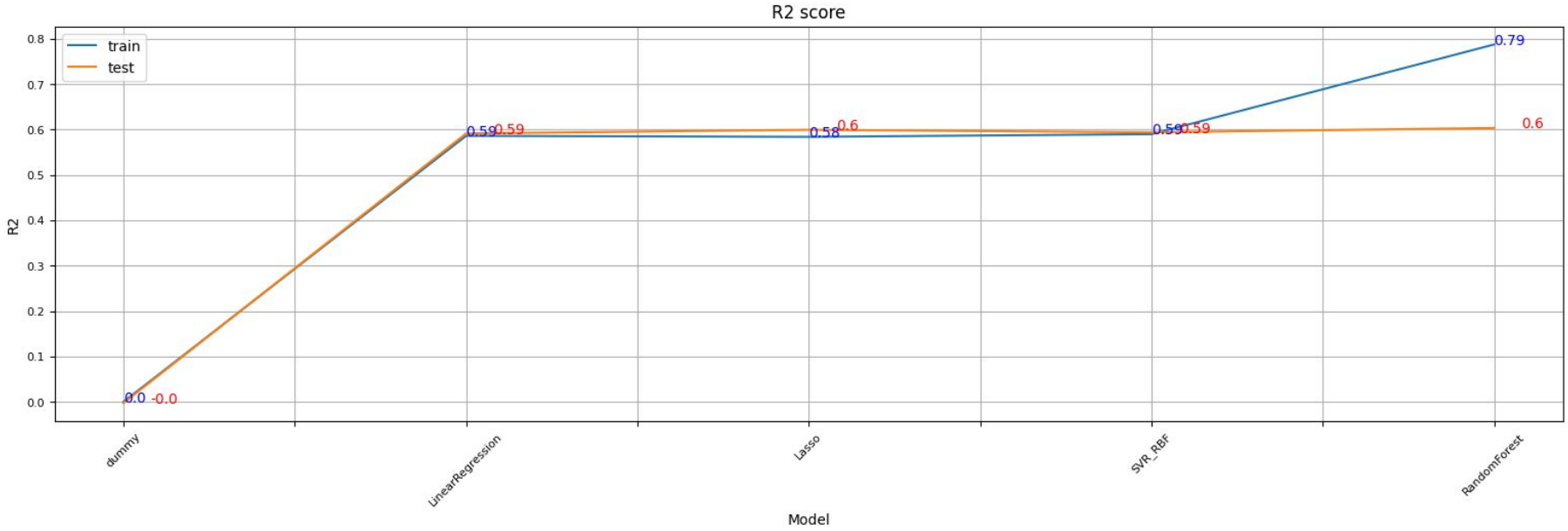


➜ **Régression Linéaire**

III. Prédiction

c. Évaluation, choix du modèle final et interprétable

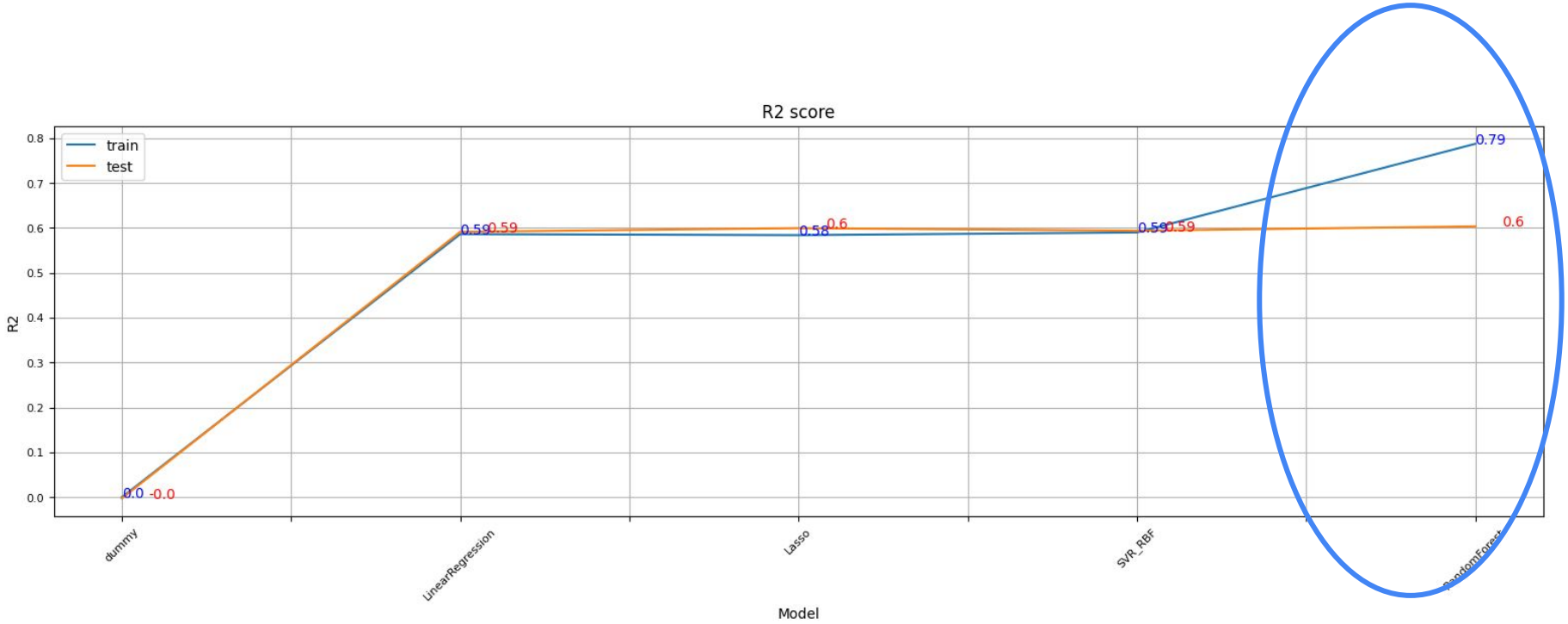
Prédiction des émissions de CO2



III. Prédiction

c. Évaluation, choix du modèle final et interprétable

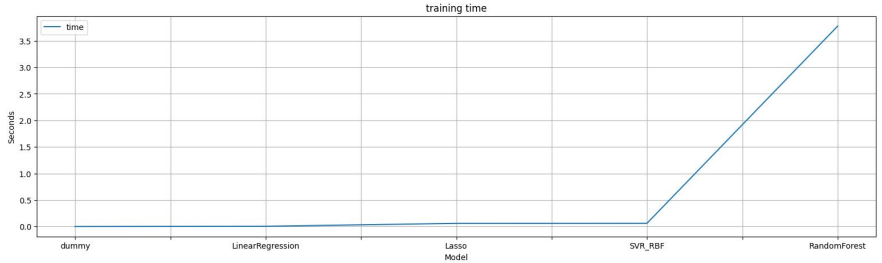
Prédiction des émissions de CO2

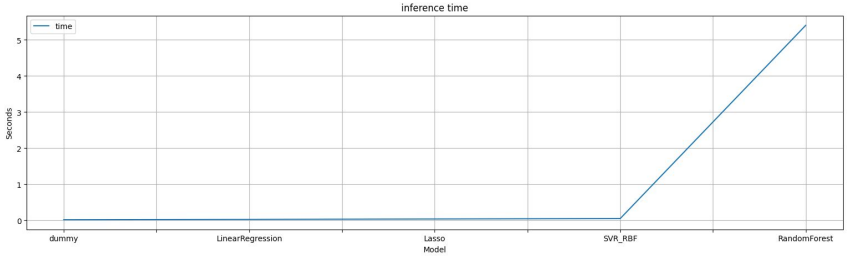


III. Prédiction

c. Évaluation, choix du modèle final et interprétable

Prédiction des émissions de CO2





➜ Régression Linéaire régularisé : Lasso

III. Prédiction

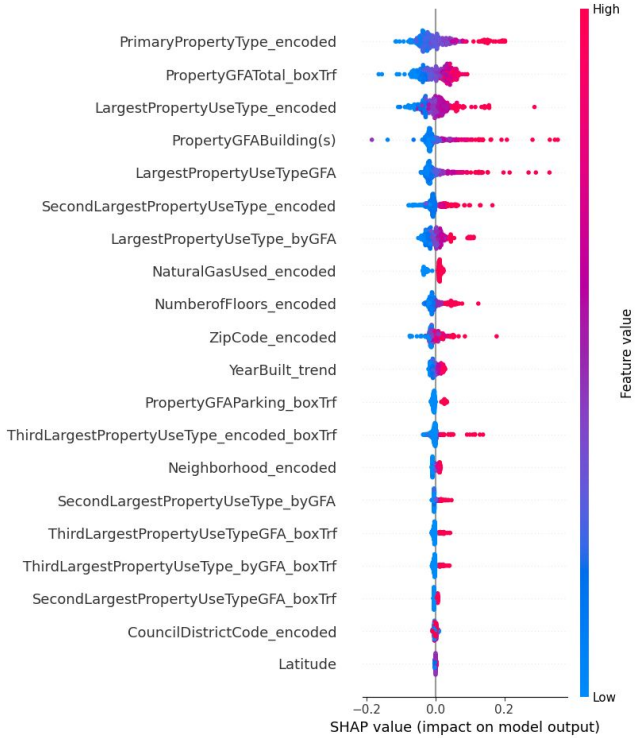
d. Interprétation des variables importantes et intégration de l'ENERGY Star score

SHAP : théorie des jeux, donnes les caractéristiques les plus importantes et leur effet sur la cible.

Pour la **consommation d'énergie, régression linéaire** :

- type propriété primaire
- plus bâtiment est plus grand, plus la consommation est plus susceptible d'être élevée.
- zipCode est plus susceptible d'avoir un effet sur la cible que le CouncilDistrictCode ou la latitude et la longitude.

l'interprétabilité difficile pour les variables categrorielles "encodées"



III. Prédiction

d. Interprétation des variables importantes et intégration de l'ENERGY Star score

Pour **émissions de carbone et Lasso** :

- NaturalGas : caractéristique la plus importante
- La taille du bâtiment, comme pour la consommation d'énergie, est également importante.
- L'année de construction est plus susceptible de contribuer à la prévision des émissions de carbone que la caractéristique a eu moins d'importance dans la prévision de la consommation d'énergie.

C02 emission avec EnergyStarScore :

R2 score sur le set d'entrainement 0.62
R2 score sur le set de test 0.64