Rapport d’étude du projet

I. Introduction

Ce rapport explore en profondeur les données fournies par Enedis pour évaluer l’impact de la classe de Diagnostic de Performance Énergétique (DPE) sur la consommation électrique des logements. Dans le cadre de ce projet, l’application **GreenTech Solutions** a été développée comme un outil d’analyse avancé pour optimiser la performance énergétique des habitations. Elle permet de prédire la consommation énergétique et de classifier chaque logement selon son étiquette DPE, offrant aux utilisateurs des informations claires et accessibles sur leur efficacité énergétique. Avec une interface intuitive, cette application facilite une prise de décision éclairée, rendant les données facilement exploitables pour des modèles de prédiction futurs.

II. Présentation des données

Les données utilisées dans ce projet proviennent de l'API d'ENEDIS et se concentrent exclusivement sur le département du Rhône (69). Deux fichiers principaux ont été extraits : `existant_69.csv` , qui contient les informations des logements anciens, et `neufs_69.csv` , qui rassemble celles des nouveaux logements.

Types de données

- **Object** : Colonnes contenant des descriptions qualitatives des biens immobiliers, telles que le type de local (maison, appartement, etc.).
- **Float** : Données numériques décimales, par exemple, la surface habitable des logements.
- **Int** : Données numériques entières, comme les codes INSEE des communes.

Ce projet s'appuie sur ces données pour évaluer l'impact du DPE sur la consommation énergétique.

III. Prétraitement des données

Le prétraitement des données est une étape cruciale dans le processus d’analyse, car il permet de transformer les données brutes en un format exploitable, assurant ainsi leur qualité et leur cohérence pour des résultats fiables.

A. Chargement et Prévisualisation des Données

Les données ont été chargées et explorées pour obtenir un aperçu général de leur structure et de leur contenu. Grâce aux bibliothèques **Pandas** et **Numpy**, nous avons importé les données depuis les fichiers CSV, puis prévisualisé les premières lignes à l'aide de `data.head()`. Cette inspection initiale a permis de vérifier la structure des colonnes, les types de données et de détecter des valeurs manquantes ou des anomalies.

B. Analyse Exploratoire des Données

Cette étape permet de comprendre la structure et les caractéristiques des données, d'identifier d'éventuels problèmes, et d'analyser la distribution de chaque variable. Afin de distinguer les logements anciens des nouveaux, une colonne "logement" a été ajoutée avec les valeurs "ancien" pour les logements existants et "neuf" pour les logements neufs. Une colonne "Année_construction" a aussi été ajoutée, avec la valeur "2024" pour les logements neufs.

1. Colonnes communes et concaténation

Une jointure a été réalisée pour permettre la prédiction à partir des deux datasets en vérifiant les colonnes communes, puis en concaténant les DataFrames `dpe_existant` et `dpe_neuf` en utilisant seulement les colonnes communes (`join='inner'` , `ignore_index=True`). Des colonnes additionnelles comme "Annee_reception_DPE", "Somme_coûts", "Coût chauffage en %" et "passoire_energetique" ont été ajoutées pour enrichir l’analyse.

2. Statistiques descriptives

Les statistiques descriptives (moyenne, écart-type, minimum, maximum et quartiles) ont été calculées pour identifier des anomalies, comme des valeurs extrêmes, et mieux comprendre la variabilité des données.

3. Valeurs manquantes

Pour identifier les colonnes nécessitant un traitement spécifique, la proportion de valeurs manquantes dans chaque colonne a été calculée. Cela a permis de déterminer si un remplacement, une imputation, ou une suppression était nécessaire avant de continuer.

4. Nettoyage des colonnes

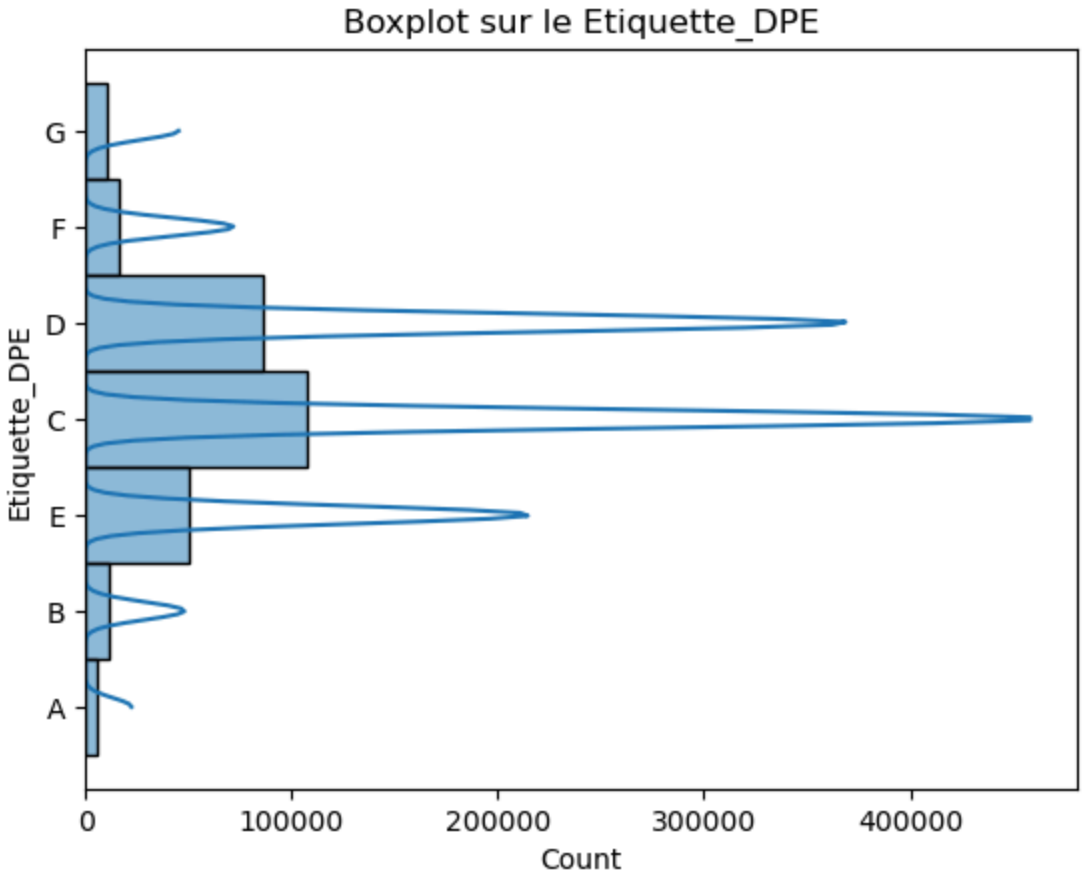
Les colonnes contenant plus de 20 % de valeurs manquantes (seuil de 0.8) ont été supprimées. Après le nettoyage, les données concaténées ont été stockées dans un fichier Excel, `data69rhone.csv` , pour les étapes suivantes de classification et de régression.

IV. Modèle de prédiction

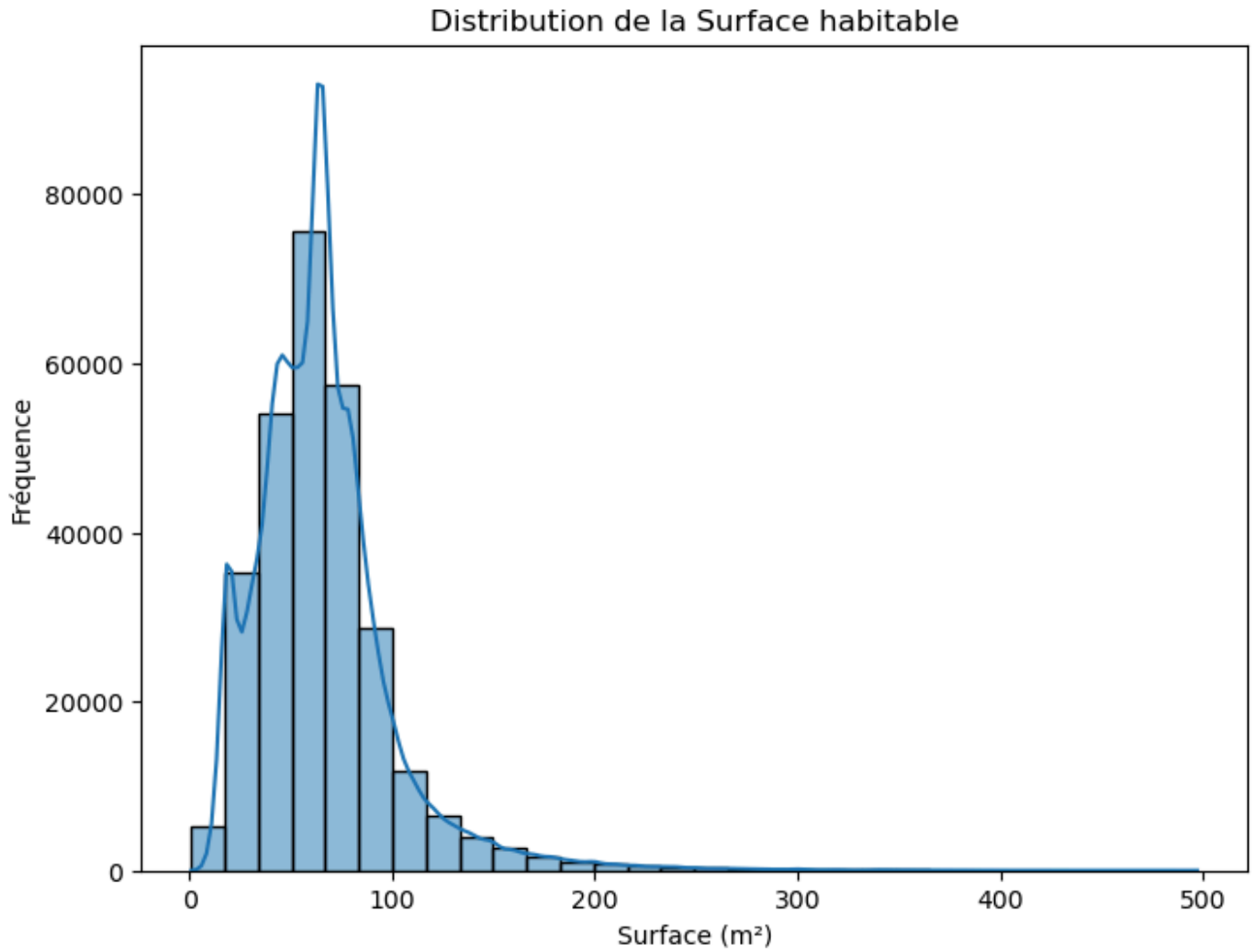
Dans cette section, nous détaillons le processus de développement des modèles de prédiction de l'étiquette DPE et de la consommation énergétique.

A. Prédiction de l'étiquette DPE

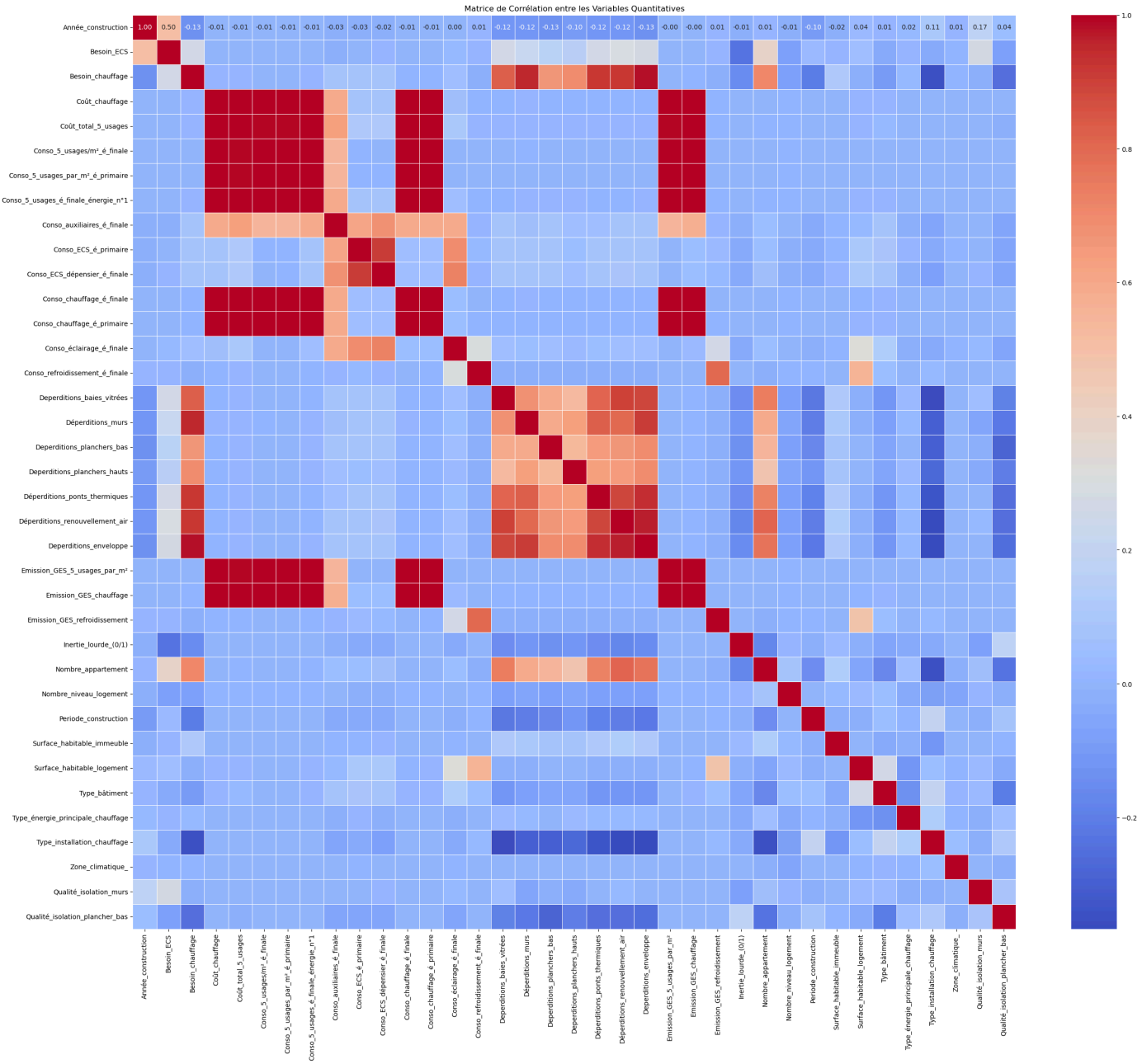
1. **Nettoyage des données de classification**
   Dans cette partie, les valeurs manquantes ont été imputées (mode pour les qualitatives, médiane pour les quantitatives), et la distribution de la variable cible, `Étiquette_DPE` , a été analysée.



   Nous avons également visualisé la distribution de la Surface habitable en m².



2. **Encodage des variables catégorielles**
   Les colonnes qualitatives et quantitatives ont été séparées, les valeurs manquantes remplacées, et les variables qualitatives encodées à l'aide de `OrdinalEncoder()` .
3. **Sélection des variables explicatives**
   Nous avons sélectionné plusieurs variables clés pour prédire l'étiquette DPE, puis nous avons analysé les corrélations afin d'identifier les variables explicatives les plus pertinentes.



4. **Échantillonnage et modèles utilisés**
   Les données ont été réparties en deux ensembles : 70 % pour l'entraînement du modèle et 30 % pour les tests.
5. **Sélection des modèles**
   Nous avons testé plusieurs modèles pour prédire l'étiquette DPE:
   -**Arbre de décision**

```
Matrice de confusion :
 [[ 1299     0   257     0     0     0     0]
 [ 1258     0  2088     0     0     0     0]
 [  238     0 30845  1199    12     1     0]
 [   58     0   188 24790   734    23     0]
 [   39     0    19    28 14615   313     0]
 [   15     0     6     1   413  4567     0]
 [   11     0    11     2    14  3106     0]]
```

```
 Rapport de classification :
               precision    recall  f1-score   support

            A       0.45      0.83      0.58      1556
            B       0.00      0.00      0.00      3346
            C       0.92      0.96      0.94     32295
            D       0.95      0.96      0.96     25793
            E       0.93      0.97      0.95     15014
            F       0.57      0.91      0.70      5002
            G       0.00      0.00      0.00      3144

     accuracy                           0.88     86150
    macro avg       0.55      0.66      0.59     86150
 weighted avg       0.83      0.88      0.86     86150
```

-**KNN**

```
Matrice de confusion :
 [[ 1369   137    40     4     5     1     0]
 [  150  2316   838    34     8     0     0]
 [   38   398 28906  2804   128    14     7]
 [    5    17  2552 20458  2545   185    31]
 [    4     6   322  3638  9997   898   149]
 [    0     1    62   540  1911  2064   424]
 [    0     4    37   139   440   717  1807]]
Rapport de classification :
               precision    recall  f1-score   support

           A       0.87      0.88      0.88      1556
           B       0.80      0.69      0.74      3346
           C       0.88      0.90      0.89     32295
           D       0.74      0.79      0.77     25793
           E       0.66      0.67      0.67     15014
           F       0.53      0.41      0.46      5002
           G       0.75      0.57      0.65      3144

    accuracy                           0.78     86150
   macro avg       0.75      0.70      0.72     86150
weighted avg       0.77      0.78      0.77     86150
```

-**Random Forest**

```
Matrice de confusion :
 [[ 1537    19     0     0     0     0     0]
 [    1  3312    33     0     0     0     0]
 [    1     5 32085   199     4     1     0]
 [    0     0   163 25449   178     3     0]
 [    0     0    10   180 14707   110     7]
 [    0     0     2     9    99  4848    44]
 [    0     0     0     2    14    53  3075]]
Rapport de classification :
              precision    recall  f1-score   support

           A       1.00      0.99      0.99      1556
           B       0.99      0.99      0.99      3346
           C       0.99      0.99      0.99     32295
           D       0.98      0.99      0.99     25793
           E       0.98      0.98      0.98     15014
           F       0.97      0.97      0.97      5002
           G       0.98      0.98      0.98      3144

    accuracy                           0.99     86150
   macro avg       0.99      0.98      0.98     86150
weighted avg       0.99      0.99      0.99     86150
```

-**KNN over sampling**

```
Matrice de confusion :
 [[ 1384   145    17     7     1     1     1]
 [  182  2753   358    40    10     3     0]
 [   52  1158 27638  3065   309    53    20]
 [    8    65  1959 19534  3616   500   111]
 [    2    15   196  2490 10131  1837   343]
 [    0     1    27   280  1234  2734   726]
 [    0     4    12    65   257   743  2063]]
Rapport de classification :
               precision    recall  f1-score   support

           A       0.85      0.89      0.87      1556
           B       0.66      0.82      0.74      3346
           C       0.91      0.86      0.88     32295
           D       0.77      0.76      0.76     25793
           E       0.65      0.67      0.66     15014
           F       0.47      0.55      0.50      5002
           G       0.63      0.66      0.64      3144

    accuracy                           0.77     86150
   macro avg       0.71      0.74      0.72     86150
weighted avg       0.78      0.77      0.77     86150
```

-**Regression logistique**

```
Matrice de confusion :
 [[   93     0  1330   132     1     0     0]
 [   36     0  2897   389    18     5     1]
 [   17     0 24789  7214   205    59    11]
 [    1     0  5626 18326  1521   250    69]
 [    0     0  1231 10337  3004   351    91]
 [    0     0   267  2984  1400   294    57]
 [    0     0   150  1405  1090   446    53]]
```

```
                precision    recall  f1-score   support

           A       0.63      0.06      0.11      1556
           B       0.00      0.00      0.00      3346
           C       0.68      0.77      0.72     32295
           D       0.45      0.71      0.55     25793
           E       0.41      0.20      0.27     15014
           F       0.21      0.06      0.09      5002
           G       0.19      0.02      0.03      3144

    accuracy                           0.54     86150
   macro avg       0.37      0.26      0.25     86150
weighted avg       0.49      0.54      0.49     86150
```

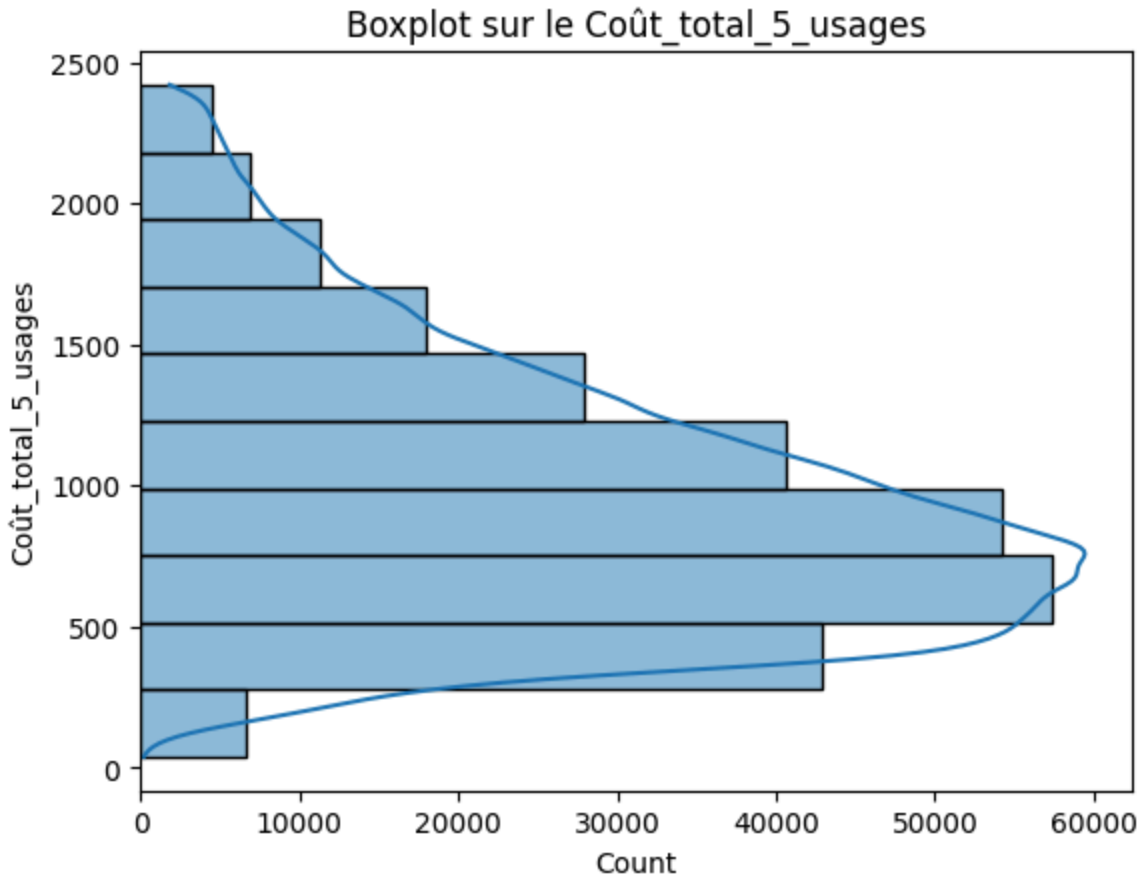
-Xgboost

```
Matrice de confusion :
 [[ 1554     2     0     0     0     0     0]
 [    2  3322    22     0     0     0     0]
 [    2    10 32118   158     7     0     0]
 [    0     0   174 25470   143     6     0]
 [    0     0     6   188 14705   109     6]
 [    0     0     1     7   130  4807    57]
 [    0     0     0     2     3    75  3064]]
Rapport de classification :
              precision    recall  f1-score   support

           A       1.00      1.00      1.00      1556
           B       1.00      0.99      0.99      3346
           C       0.99      0.99      0.99     32295
           D       0.99      0.99      0.99     25793
           E       0.98      0.98      0.98     15014
           F       0.96      0.96      0.96      5002
           G       0.98      0.97      0.98      3144

    accuracy                           0.99     86150
   macro avg       0.99      0.98      0.98     86150
weighted avg       0.99      0.99      0.99     86150
```

1. **Modèle sélectionné et variables retenues**
   Le modèle **Random Forest** a atteint une précision élevée, avec une matrice de corrélation indiquant une meilleure performance dans la prédiction des données. Pour optimiser la prédiction, nous avons sélectionné les 15 variables les plus pertinentes.
   (capture)

B. Prédiction de la consommation

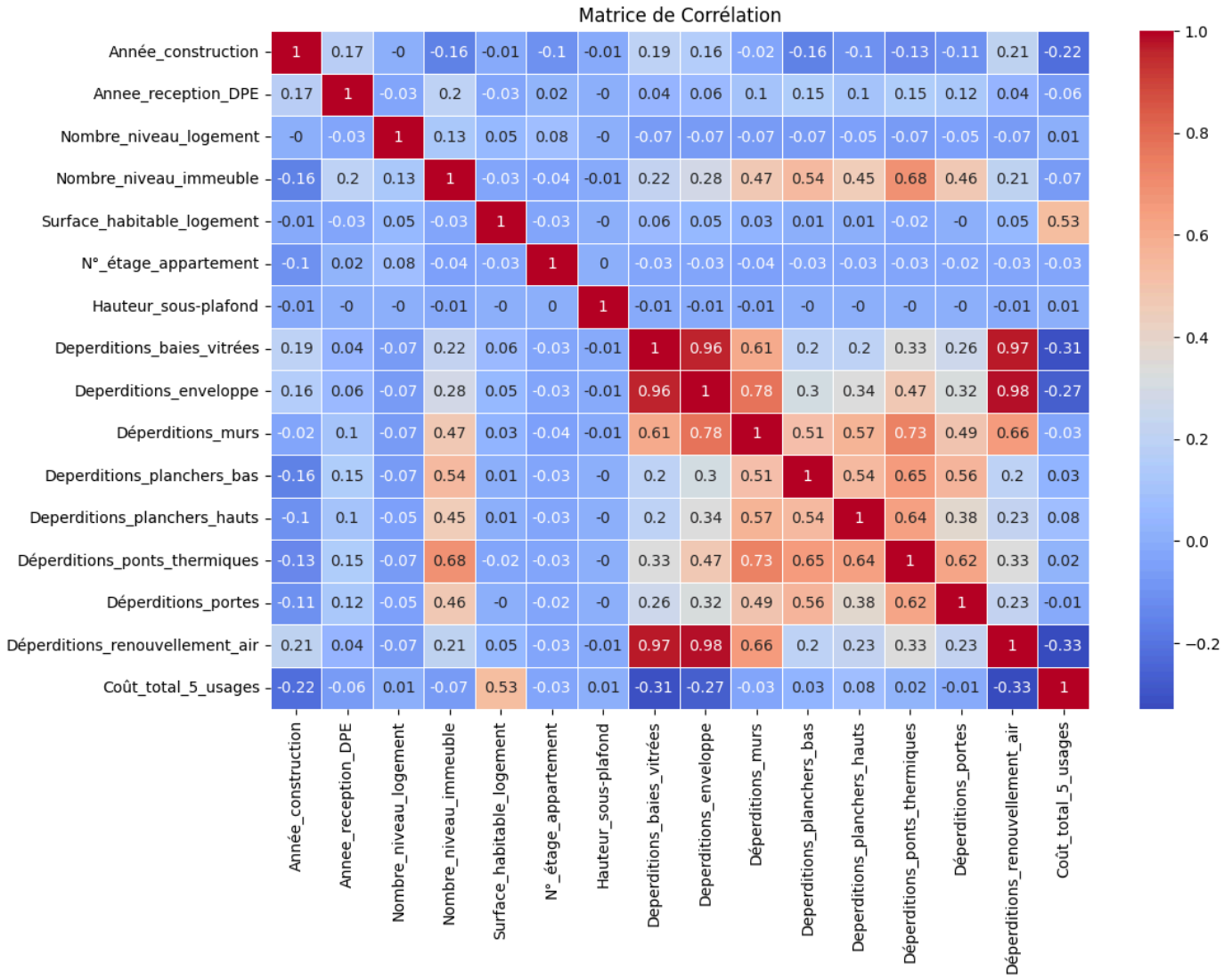
1. **Nettoyage des données de régression**
   La première étape dans l'élaboration de notre modèle a été le nettoyage des données cibles, c'est-à-dire les consommations. Ce nettoyage s'est fait en deux étapes :
   - Suppression des valeurs manquantes
   - Élimination des valeurs situées en dehors des 15^{e} et 95^{e} percentiles pour éviter les valeurs extrêmes qui pourraient biaiser le modèle.



2. **Normalisation des données numériques**
   Nos données numériques étant exprimées dans des unités différentes, nous avons procédé à une normalisation pour ramener toutes les valeurs à la même échelle.
3. **Encodage des variables catégorielles**
   Nous avons utilisé le label encoding pour transformer les variables catégorielles, facilitant ainsi leur exploitation dans le modèle.

4. **Sélection des variables explicatives**

Nous avons utilisé la corrélation pour sélectionner les variables explicatives pertinentes.

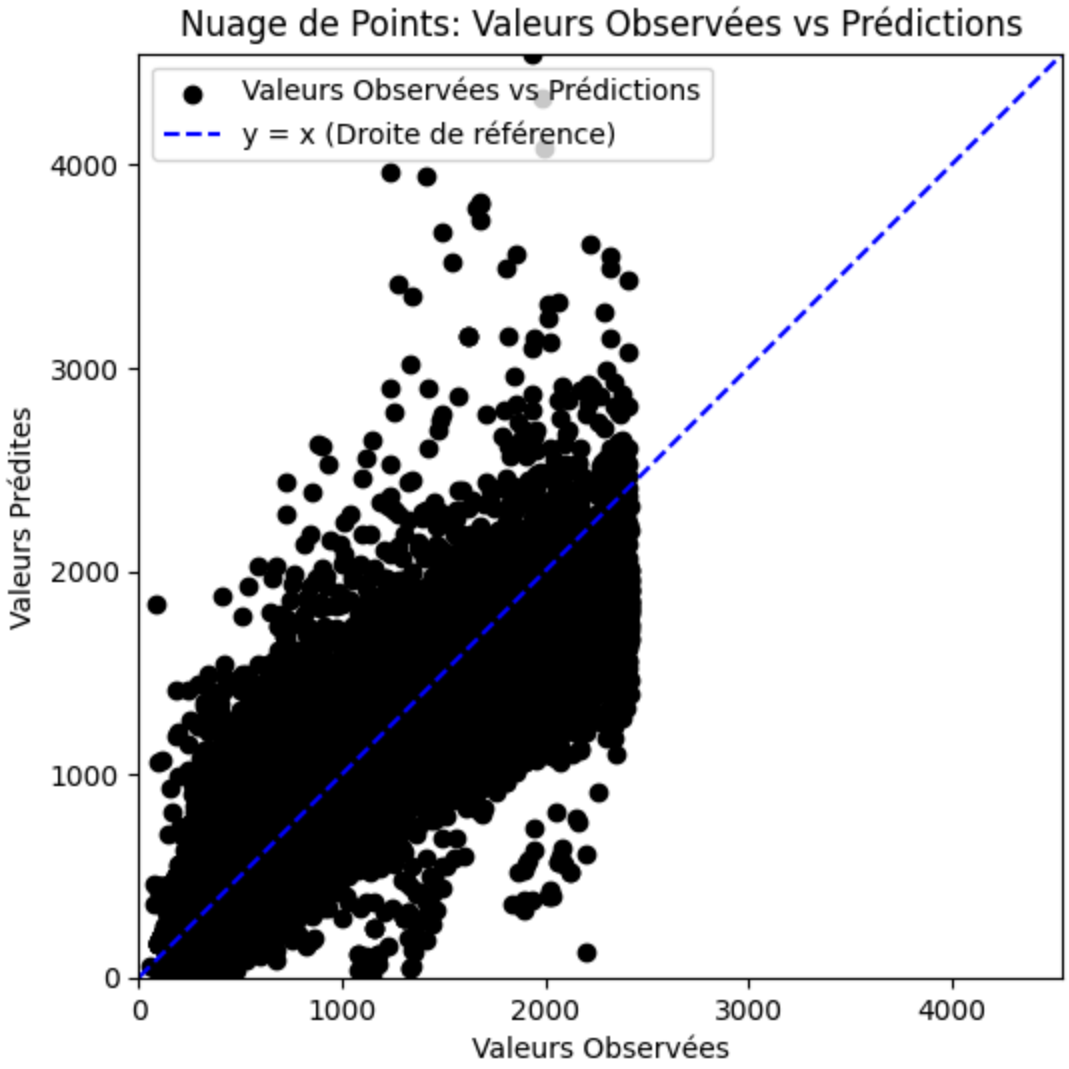


5. **Sélection des modèles**

Nous avons testé plusieurs modèles pour prédire la consommation :

- **Régression linéaire**

  Le premier modèle testé a été la régression linéaire, qui nous a donné une **RMSE de 218**.

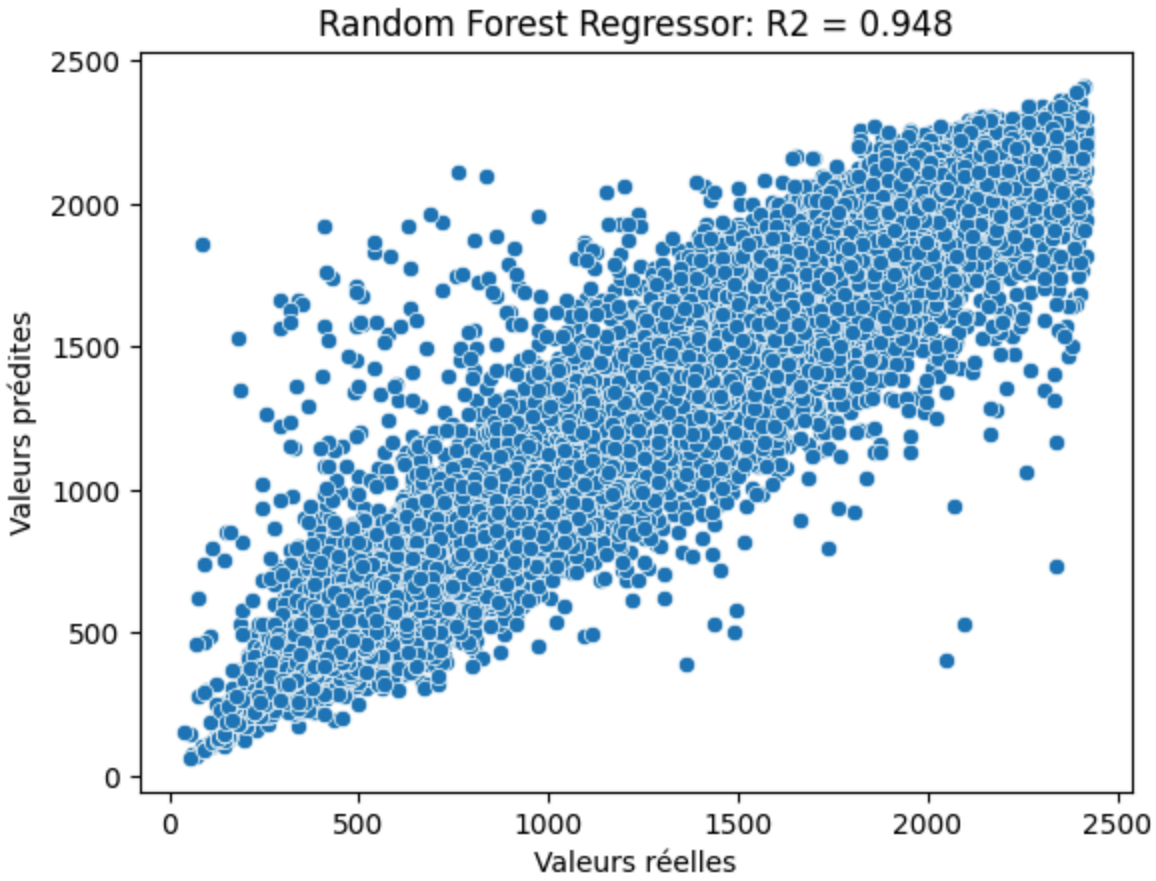



```
MAE : 159.16537949100604
RMSE : 218.0557770316835
R² : 0.7861797976715609
```

- **Arbre de décision**


```
R2 : 0.8997359649514172
RMSE : 0.315864033365754
MAE : 0.17038447658454858
```

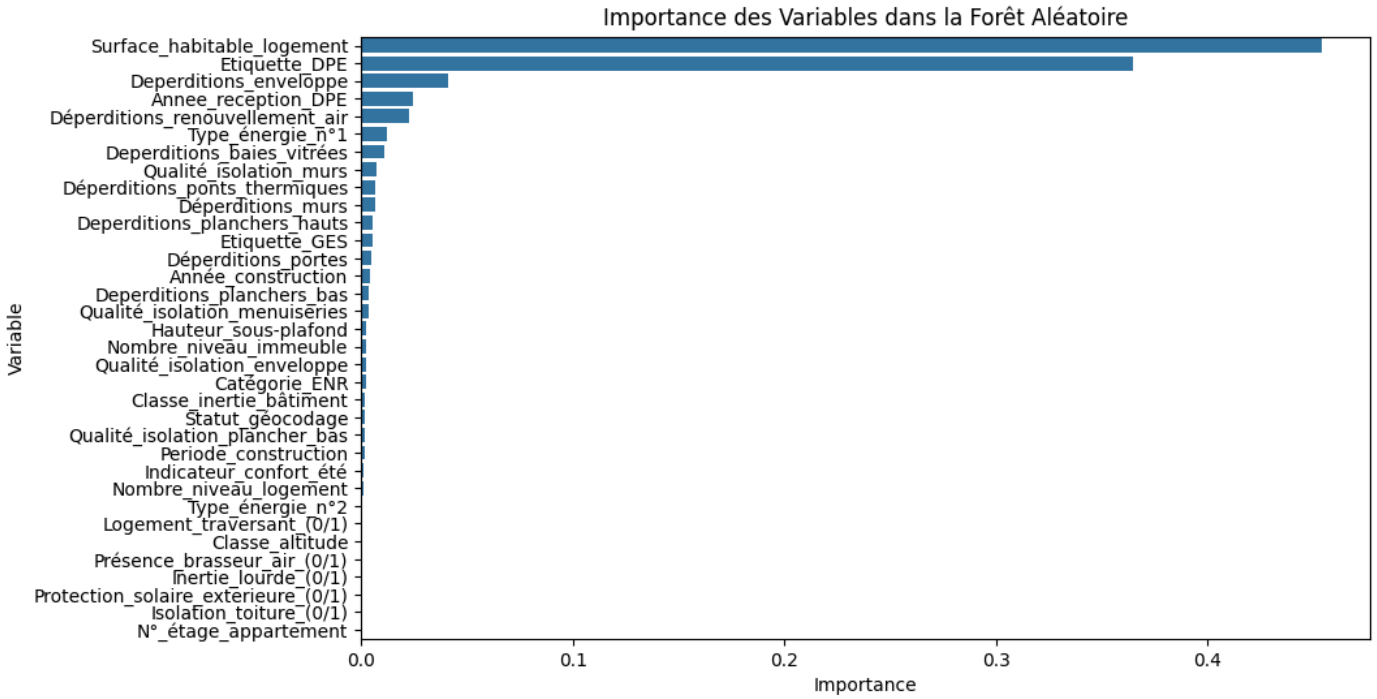
- **Random Forest Regressor**




```
RMSE : 107.66837655059616
MAE : 61.238307774082074
R2 : 0.9478697423492155
```

6. **Modèle sélectionné et variables retenues**

   Au vu des scores des différents modèles, nous avons opté pour le Random Forest. Nous avons également sélectionné les 10 variables les plus pertinentes pour optimiser la prédiction.



VI. Conclusions et Recommandations

L’analyse révèle que certaines variables influencent fortement l’étiquette DPE et la consommation énergétique. Le **RandomForestClassifier** et le **RandomForestRegressor** ont offert les meilleures précisions. Les recommandations incluent l’optimisation des dépenses énergétiques et l’adoption de politiques de rénovation. Parmi les limitations figurent la qualité des données et les choix de modèles. Les améliorations possibles incluent l'exploration de nouveaux algorithmes et un ajustement plus poussé des hyperparamètres.