OPENCLASSROOMS CentraleSupélec

Anticipez les besoins en consommation électrique de bâtiments

Formation Data Scientist - Janvier / Novembre 2022

Auteur: Eric TREGOAT

Mentor: Benjamin TARDY

Evaluateur: Eric NANA NJOYA

Ordre du jour de la soutenance

- **Présentation (25 minutes)**
  - Présentation de la problématique, de son interprétation et des pistes de recherche envisagées
  - Présentation du cleaning effectué, du feature engineering et de l'exploration
  - Présentation des différentes pistes de modélisation effectuées
  - Présentation du modèle final sélectionné ainsi que des améliorations effectuées
- **Discussion**
- **Débriefing**

Problématique, interprétation et pistes de recherche envisagées

❑ Problématique

- Pour atteindre son objectif de ville neutre en émissions de carbone en 2050, nous sommes mandatés par la ville de Seattle pour nous intéresser aux émissions des bâtiments non destinés à l'habitation.
- A partir de mesures effectuées sur un échantillon de bâtiments, nous voulons prédire les émissions de GES et la consommation totale d'énergie de bâtiments pour lesquels il n'y pas encore de mesure effectuée.
- Les prédictions sont à baser sur les données déclaratives du permis d'exploitation commerciale (caractéristiques et usage des bâtiments).
- Nous recherchons également à évaluer l'intérêt de l'"ENERGY STAR Score" pour la prédiction d'émissions, qui est fastidieux à calculer avec l'approche utilisée actuellement.

❑ Interprétation et pistes de recherche envisagées

1. Recherche de modélisation des émissions de GES et de la consommation totale d'énergie à partir des données de l'échantillon en utilisant le machine learning (apprentissage supervisé).
2. Examen de l'importance de l'"ENERGY STAR Score" sur la prédiction des émissions de GES pour en évaluer l'intérêt, en examinant la performance de la modélisation avec ou sans cette donnée, et, le cas échéant, le poids de cette donnée dans la modélisation.
3. Conclusion sur la possibilité de généralisation en fonction du niveau de performance de la modélisation et de la nécessité d'utilisation ou pas de l'"ENERGY STAR Score".

Cleaning effectué, feature engineering et exploration (1/8)

Démarche :

1. **Prise de connaissance** des jeux de données de 2015 et 2016
2. **Mise en cohérence** et concaténation des 2 jeux de données
3. **Réduction** du jeu aux données et variables utiles au projet
4. **Examen** des valeurs aberrantes et autres contrôles de cohérence
5. **Traitement** des valeurs manquantes
6. **Analyses exploratoires univariées** pour la préparation des données au machine learning
7. **Analyses exploratoires multivariées** pour la sélection finale du jeu de données en entrée du machine learning

Cleaning effectué, feature engineering et exploration (2/8)

Point clé : choix des variables cibles

- **Energie:** nous retenons 'SiteEnergyUse(kBtu)' et filtrons toutes les autres variables relatives à l'énergie afin d'éviter du « data leakage ».

- **Emissions de GES**: nous retenons 'TotalGHGEmissions' et filtrons l'autre variables afin d'éviter du « data leakage ».

Cleaning effectué, feature engineering et exploration (3/8)

Point clé : filtrage des bâtiments ‘résidentiels’ vs ‘non résidentiels’

- **Filtrage** des bâtiments exclusivement destinés au résidentiel.
- Conservation des bâtiments **mixtes** ‘résidentiel’ et ‘non résidentiel’ pour maximiser la taille de l’échantillon et ne pas négliger certaines catégories d’utilisation particulièrement représentées pour ces bâtiments (ex: retail).
- Un **second filtrage** est effectué lors de la simplification des catégories de type et usage des bâtiments, les catégories relatives au résidentiel étant placées dans la catégorie ‘Other’.

Cleaning effectué, feature engineering et exploration (4/8)

Point clé : correction des données de surface d'utilisation

- Les 3 données de surface totale ('PropertyGFAParking', 'PropertyGFABuilding(s)' et 'PropertyGFATotal') sont parfaitement cohérentes entre elles et sont considérées comme la référence fiable.

- Les 3 données de surface d'utilisation ('LargestPropertyUseTypeGFA', 'SecondLargestPropertyUseTypeGFA', 'ThirdLargestPropertyUseTypeGFA') présentent des erreurs et incohérences qui sont corrigées en s'appuyant sur les données de surface totale.

- Au total:
  - la somme des 3 surfaces d'utilisation est égale à la surface totale de la propriété ;
  - La surface d'utilisation de parking est la même que celle de la surface totale d'utilisation de parking.

Cleaning effectué, feature engineering et exploration (5/8)

Point clé : transformation des variables catégorielles

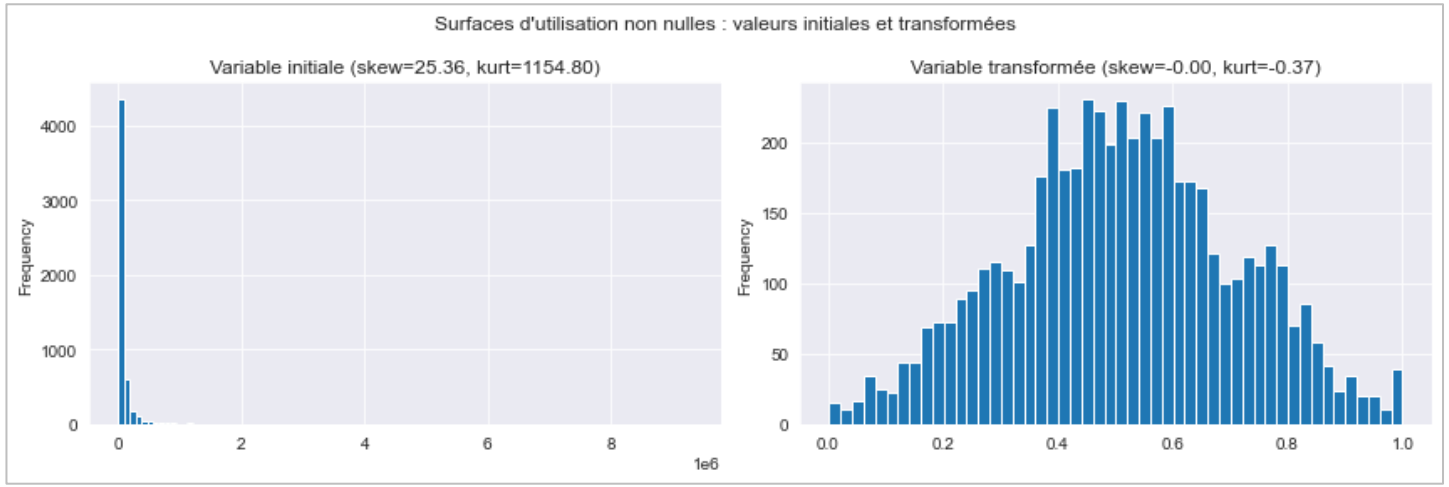
- Les variables catégorielles de type et utilisation des bâtiments sont réduites à 16 catégories:
  - Mêmes catégories pour le type et l'utilisation
  - Une catégorie 'Other' qui représente 23% des types de bâtiment et 13% de leurs utilisations.

- Les variables catégorielles sont transformées en autant de variables numériques que de catégories :
  - Type de bâtiment: 0 ou 1 pour chaque catégorie de type de bâtiment
  - Utilisation de bâtiment: valeur de la surface d'utilisation

Cleaning effectué, feature engineering et exploration (6/8)

Point clé : transformation et normalisation des features et targets

- S'appuie sur l'analyse univariée
- Toutes les features et targets sont normalisées entre 0 et 1 (MinMax)
- Avant cette normalisation, certaines variables sont transformées afin d'améliorer leur distribution:
  - Variables de surface d'utilisation: ramenées au rapport de la surface totale maximum des propriétés puis transformées avec log(x+eps) – log(eps), avec eps optimisé pour skew=0



  - Nombre d'étages, énergie et émission de GES: transformation log
  - Année de construction: ramenée à l'âge du bâtiment

Cleaning effectué, feature engineering et exploration (7/8)

Point clé : filtrage des features de type de bâtiment

- S'appuie sur l'analyse multivariée:
  - Forte corrélation entre les variables de même catégorie de type et d'utilisation de bâtiment
  - Aucune corrélation entre les variables de catégories différentes de type et d'utilisation de bâtiment

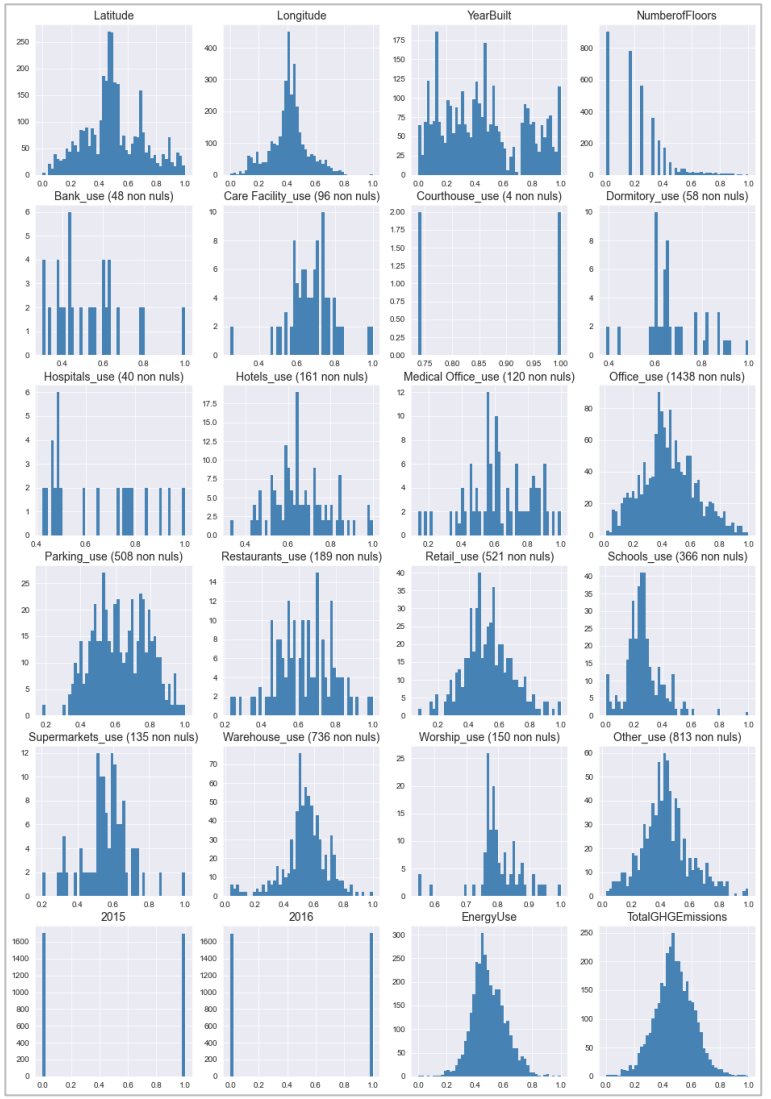
- Filtrage des variables de catégorie de type de bâtiment

	Care Facility_use	Dormitory_use	Hotels_use	Medical Office_use	Office_use	Other_use	Retail_use	Schools_use	Supermarkets_use	Warehouse_use	Worship_use
Care Facility_type	0.91	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.00	0.01	0.00
Dormitory_type	0.00	0.70	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Hotels_type	0.00	0.00	0.90	0.00	0.03	0.01	0.00	0.00	0.00	0.01	0.00
Medical Office_type	0.00	0.00	0.00	0.80	0.01	0.00	0.00	0.00	0.00	0.01	0.00
Office_type	0.01	0.01	0.02	0.01	0.70	0.03	0.00	0.03	0.01	0.05	0.02
Other_type	0.00	0.00	0.01	0.00	0.04	0.49	0.00	0.02	0.00	0.03	0.01
Retail_type	0.00	0.00	0.00	0.00	0.02	0.01	0.45	0.01	0.00	0.01	0.00
Schools_type	0.00	0.00	0.00	0.00	0.06	0.03	0.02	0.83	0.00	0.02	0.00
Supermarkets_type	0.00	0.00	0.00	0.00	0.01	0.01	0.00	0.00	0.56	0.01	0.00
Warehouse_type	0.01	0.00	0.01	0.01	0.04	0.04	0.02	0.02	0.01	0.80	0.01
Worship_type	0.00	0.00	0.00	0.00	0.02	0.01	0.01	0.00	0.00	0.01	0.95

Cleaning effectué, feature engineering et exploration (8/8)

Jeu de données pour le machine learning :

- Echantillon de **3396 bâtiments.**
- **2 variables cibles** (Energie et émission GES)
- **23 features**, dont une feature à examiner spécifiquement (EnergyStarScore) avec seulement 2280 valeurs renseignées.
- **2 variables d'identification** des bâtiments, permettant en particulier d'assurer la correspondance entre le jeu de données initial et le jeu préparé pour le machine learning.

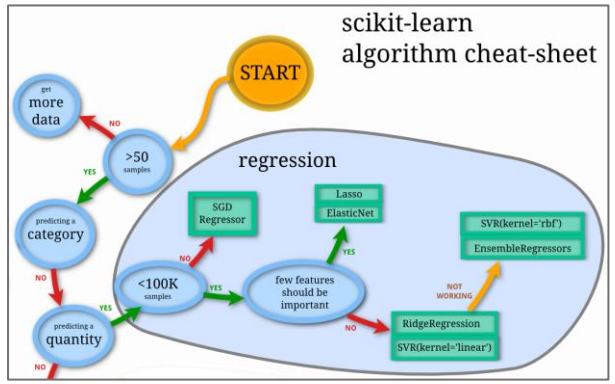


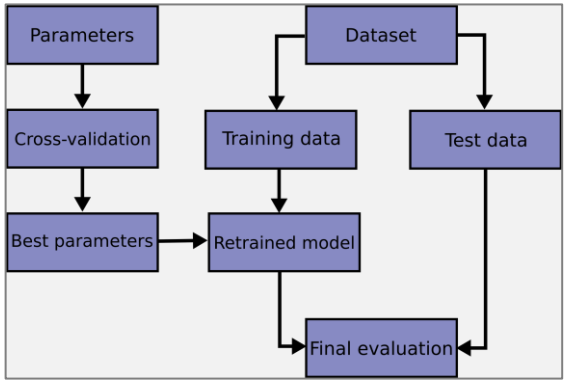
OPENCLASSROOMS
CentraleSupélec

Modélisations effectuées (1/6)

Démarche :

1. Objectif de la modélisation, cibles, features
2. Type de modélisation: régression
3. Liste des modèles à tester:
   - Baseline: DummyRegressor
   - Modèles de régression linéaires (Ridge, Lasso et ElasticNet)
   - Modèles non linéaires (SVR, kRR et MLP)
   - Modèles ensemblistes (Random Forest, AdaBoost et XGBoost)





4. Processus d'évaluation:
   - Découpe du jeu: entrainement et test
   - Métriques d'évaluation: MSE (+ r2, MAE, MedAE) et temps d'entrainement unitaire
   - Recherche sur grille avec validation croisée (CV)
   - Conditions identiques et répétables de comparaison (CV=4, random_state=0, n_jobs=-1)
   - Comparaison et enregistrement des résultats
5. Fonctions support au processus d'évaluation
   - Fonction d'entrainement et optimisation en fonction de plages d'hyperparamètres
   - Visualisation de l'influence combinée des paramètres
   - Visualisation de l'importance des features
   - Courbes d'apprentissage
   - Evaluation comparative des résultats
6. Sélection et améliorations
7. Bilan et conclusion

Modélisations effectuées (2/6)

Point clé : entrainement et optimisation fonction de plages d'hyperparamètres

Code récurrent pour chaque modélisation, avec adaptation du nom de modèle et nature des hyperparamètres:

Spécifique modèle

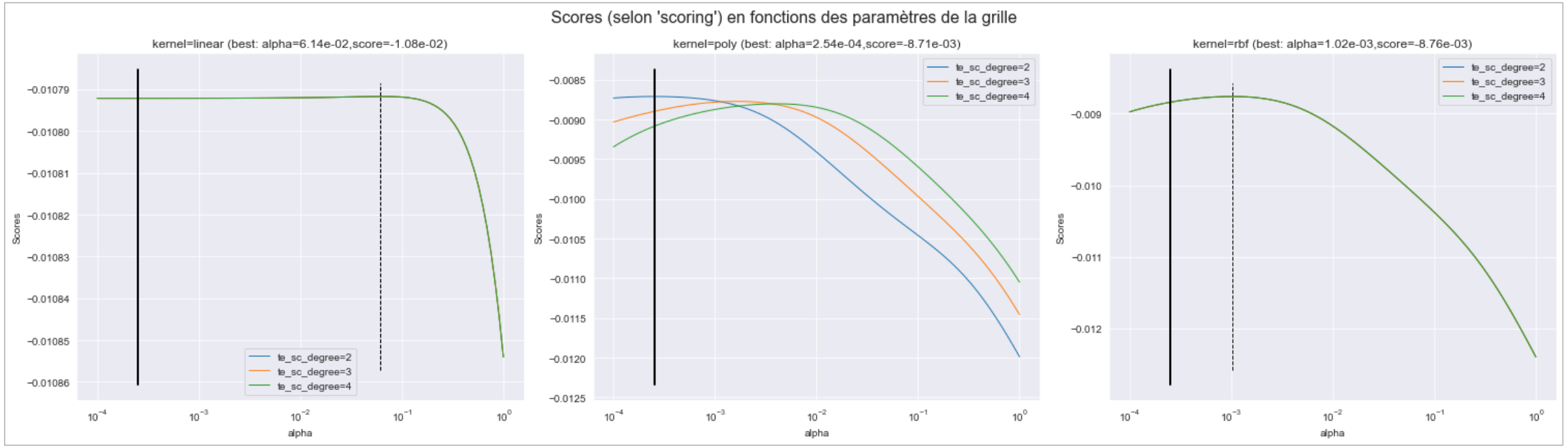
```
from sklearn.ensemble import RandomForestRegressor
model, model_name = fct.append_model(RandomForestRegressor(random_state=rs))

# Définition des grilles de recherche
param_grid[targets[0]] = {'n_estimators': np.array(range(100, 200, 10)),
                          'max_features': [14, 15]}
param_grid[targets[1]] = {'n_estimators': np.array(range(300, 400, 10)),
                          'max_features': [15, 16, 17]}
param_grid[targets[2]] = {'n_estimators': np.array(range(200, 300, 10)),
                          'max_features': [12, 14, 15]}
```

Générique

```
gr = []
for idx in range(3):
    print("\n\n", Fore.BLACK + Style.BRIGHT + Back.WHITE + f"Modèle : {model_name} - {targets[idx]}\n" + Style.RESET_ALL)
    gr.append(GridSearchCV(model, param_grid[targets[idx]], scoring=Scoring, cv=CV, return_train_score=True))
    gr[idx] = fct.search_best_model(gr[idx], targets[idx], Xtr_list[idx], ytr_list[idx])
    fct.plt_grid(gr[idx], param_grid[targets[idx]], sort=True, x='n_estimators', scale='linear')
    fct.plot_feature_importance(gr[idx].best_estimator_, f_names[idx])
    fct.model_eval(gr[idx].best_estimator_, model_name, targets[idx], Xtr_list[idx], ytr_list[idx], Xte_list[idx], yte_list[idx])
    fct. learning_graph(gr[idx].best_estimator_, Xtr_list[idx], ytr_list[idx], scoring=Scoring, random_state=rs)
```

Modélisations effectuées (3/6)

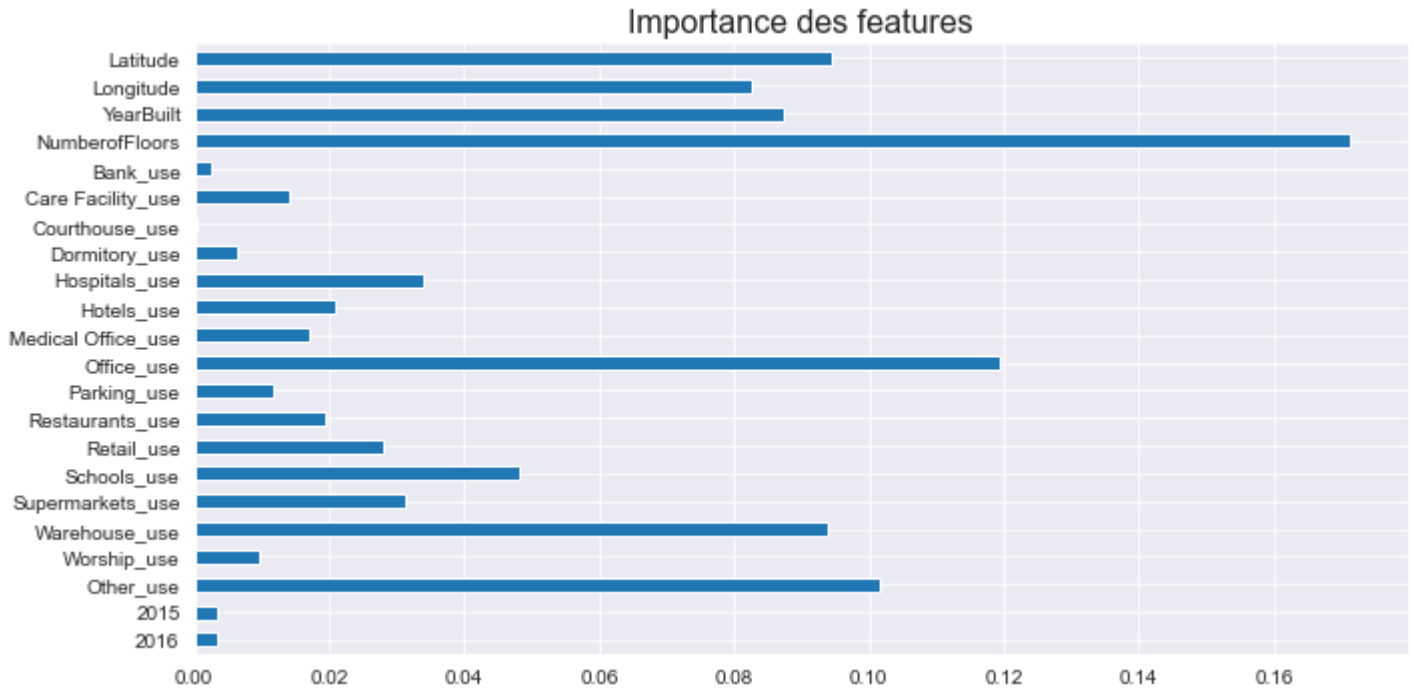
Point clé : visualisation de l'influence combinée des paramètres



OPENCLASSROOMS
CentraleSupélec

Modélisations effectuées (4/6)

Point clé : visualisation de l'importance des features

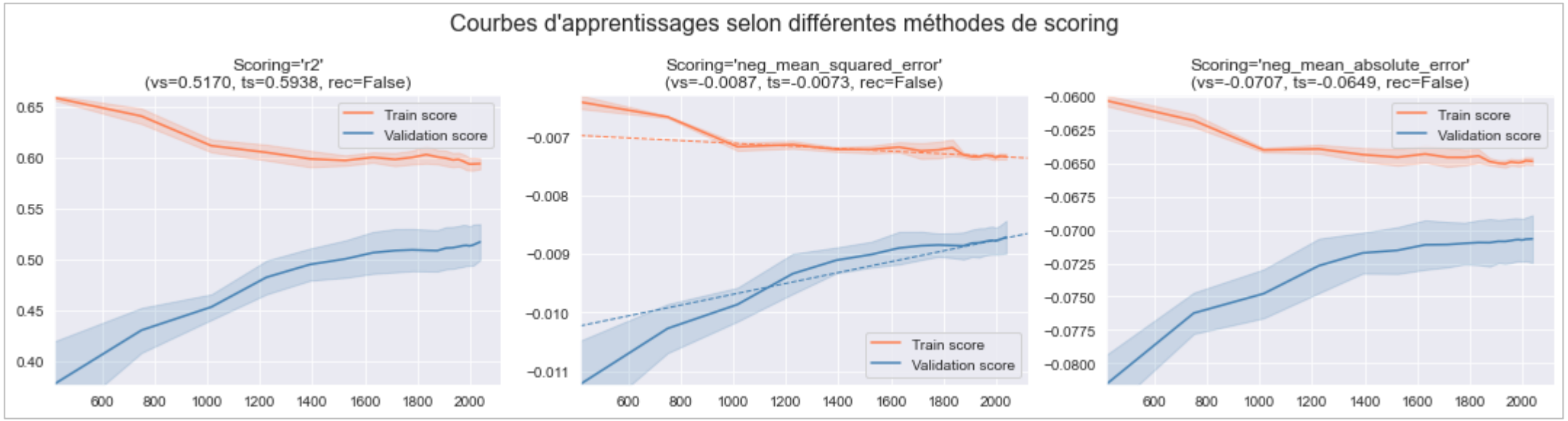


OPENCLASSROOMS
CentraleSupélec

Modélisations effectuées (5/6)

Point clé : courbes d'apprentissage

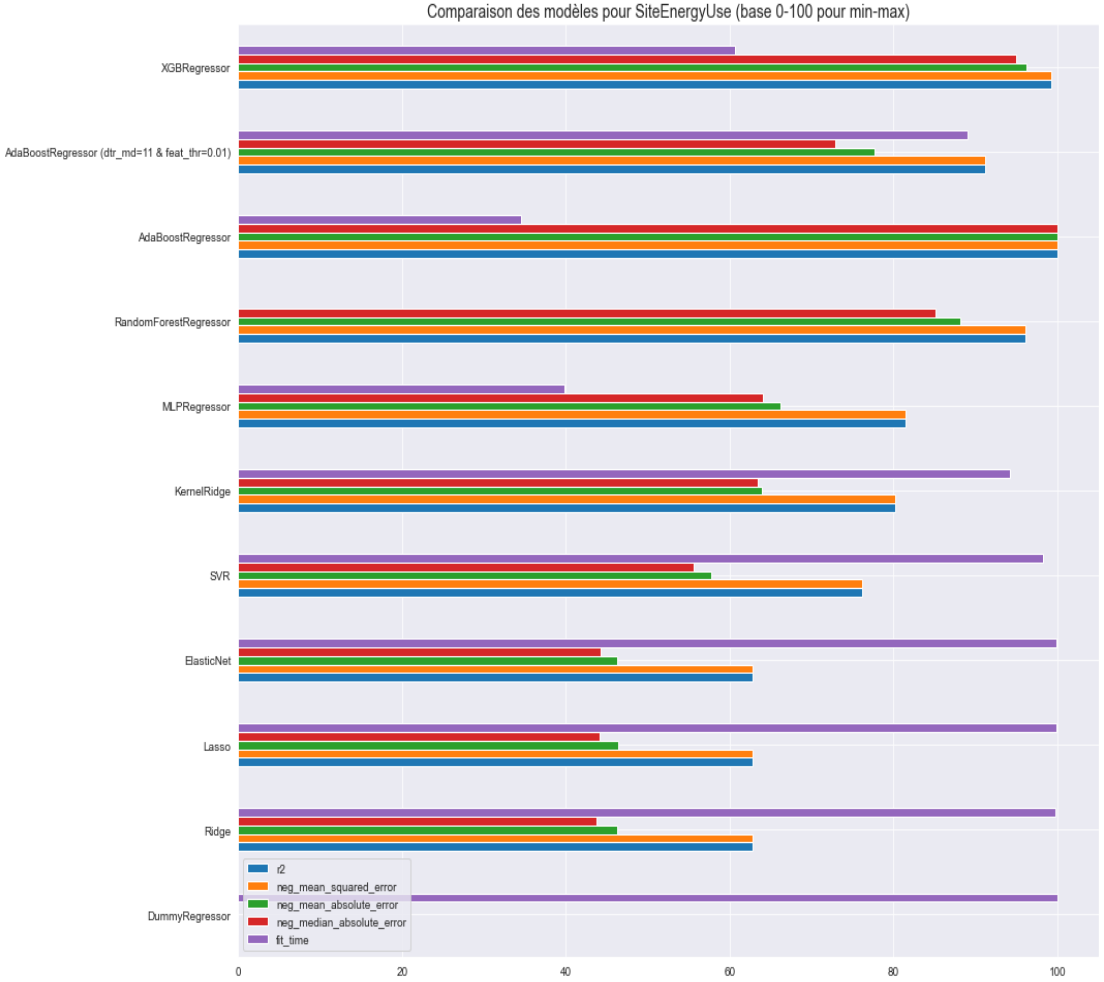
- Courbes d'apprentissage selon 3 métriques de scoring, montrant les scores d'entrainement et de validation



Modélisations effectuées (6/6)

Point clé : évaluation comparative des résultats (ex: SiteEnergyUse)

	r2	neg_mean_squared_error	neg_mean_absolute_error	neg_median_absolute_error	fit_time
DummyRegressor	-0.000	-0.014080	-0.093904	-0.080132	-0.000s
Ridge	0.572	-0.006024	-0.059136	-0.048694	-0.007s
Lasso	0.572	-0.006023	-0.059089	-0.048443	-0.004s
ElasticNet	0.572	-0.006023	-0.059097	-0.048381	-0.004s
SVR	0.694	-0.004308	-0.050524	-0.040162	-0.047s
KernelRidge	0.731	-0.003783	-0.045845	-0.034550	-0.158s
MLPRegressor	0.742	-0.003629	-0.044206	-0.034149	-1.654s
RandomForestRegressor	0.876	-0.001751	-0.027675	-0.018948	-2.746s
AdaBoostRegressor	0.912	-0.001245	-0.018762	-0.008296	-1.799s
AdaBoostRegressor (dtr_md=11 & feat_thr=0.01)	0.831	-0.002382	-0.035567	-0.027817	-0.301s
XGBRegressor	0.904	-0.001346	-0.021631	-0.011874	-1.079s



OPENCLASSROOMS CentraleSupélec

Modèle final sélectionné

Modèle sélectionné : AdaBoostRegressor

- Meilleur score, avec un niveau assurant une bonne qualité de modélisation
- Permet la modélisation des émissions de GES sans l'EnergyStarScore
- Temps d'entrainement raisonnable (1,8 à 3,7s)

Modèles:

- Consommation d'énergie:
  AdaBoostRegressor(base_estimator=DecisionTreeRegressor(max_features='auto'), learning_rate=1.5, loss='linear', n_estimators=190)
- Emissions de GES
  AdaBoostRegressor(base_estimator=DecisionTreeRegressor(max_features='auto'), learning_rate=1.3, loss='linear', n_estimators=305)

Le modèle XGBoostRegressor est une excellente alternative compte tenu de son score et surtout de son temps de calcul

Comparaison entre les modèles évalués - TotalGHGEmissions sans ENERGYSTARScore

	r2	neg_mean_squared_error	neg_mean_absolute_error	neg_median_absolute_error	fit_time
DummyRegressor	-0.000	-0.019623	-0.109815	-0.090987	-0.000s
Ridge	0.429	-0.011207	-0.082840	-0.064355	-0.012s
Lasso	0.429	-0.011205	-0.082821	-0.064490	-0.003s
ElasticNet	0.429	-0.011205	-0.082823	-0.064480	-0.003s
SVR	0.532	-0.009190	-0.075899	-0.065946	-0.075s
KernelRidge	0.561	-0.008603	-0.071676	-0.057262	-0.162s
MLPRegressor	0.585	-0.008151	-0.069072	-0.053011	-1.659s
RandomForestRegressor	0.806	-0.003810	-0.043037	-0.031012	-1.914s
AdaBoostRegressor	0.851	-0.002926	-0.027643	-0.009003	-3.708s
AdaBoostRegressor (dtr_md=11 & feat_thr=0.01)	0.756	-0.004789	-0.052017	-0.043070	-0.438s
XGBRegressor	0.831	-0.003307	-0.034599	-0.018363	-0.831s

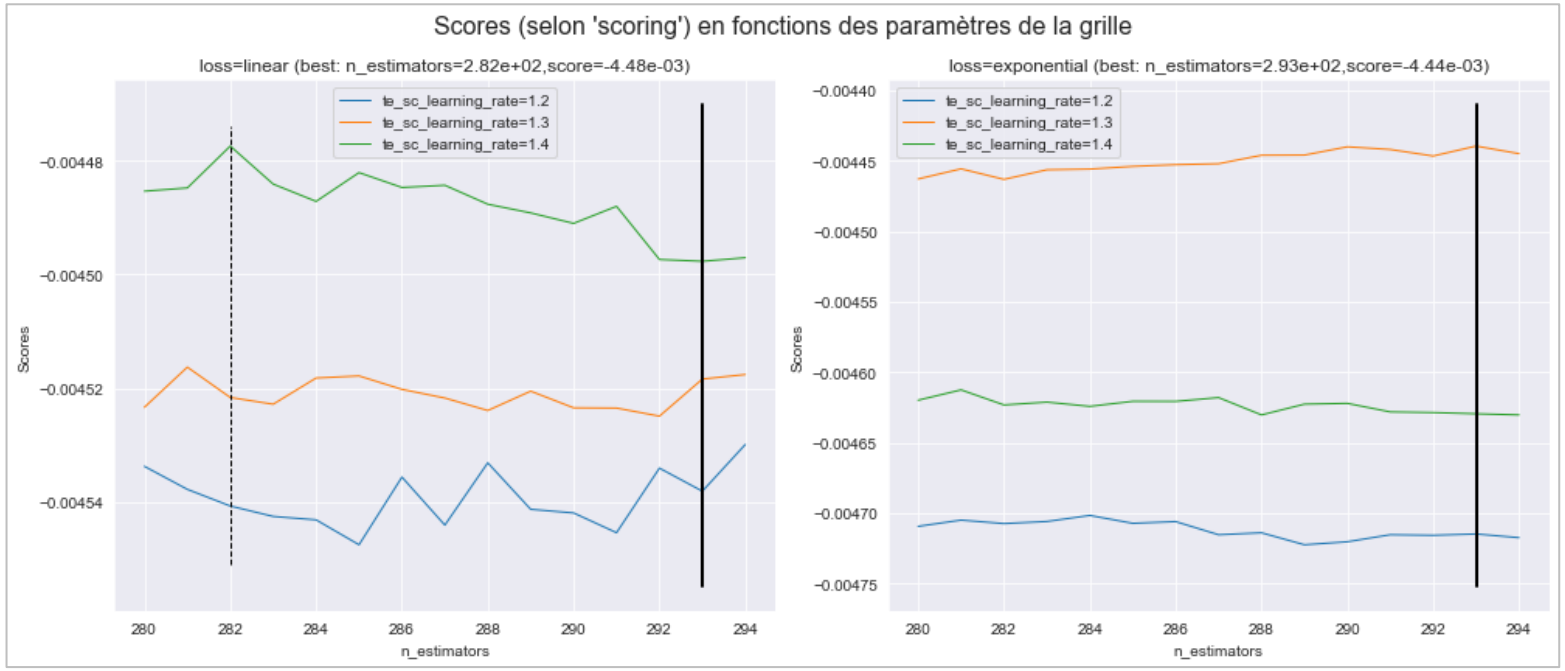
Comparaison entre les modèles évalués - TotalGHGEmissions avec ENERGYSTARScore

	r2	neg_mean_squared_error	neg_mean_absolute_error	neg_median_absolute_error	fit_time
DummyRegressor	-0.000	-0.016539	-0.098978	-0.082470	-0.000s
Ridge	0.446	-0.009167	-0.071385	-0.053559	-0.008s
Lasso	0.446	-0.009167	-0.071370	-0.054095	-0.003s
ElasticNet	0.446	-0.009166	-0.071370	-0.054077	-0.003s
SVR	0.536	-0.007668	-0.070718	-0.060639	-0.198s
KernelRidge	0.541	-0.007594	-0.063546	-0.052509	-0.059s
MLPRegressor	0.576	-0.007008	-0.060836	-0.046497	-1.058s
RandomForestRegressor	0.783	-0.003586	-0.041338	-0.029256	-0.922s
AdaBoostRegressor	0.840	-0.002645	-0.026234	-0.009537	-2.092s
XGBRegressor	0.838	-0.002681	-0.030199	-0.014689	-0.529s

Améliorations effectuées (1/2)

Optimisation des paramètres pour maximiser le score:

1. Le choix de l'apprenant faible a été déterminant: DecisionTreeRegressor(max_depth='auto')
2. Recherche combinée large des 3 hyperparamètres: base_estimator, learning_rate et n_estimators
3. Examen des courbes de score (ici MSE) et de leurs tendances pour resserrer les plages d'hyperparamètre
   → l'augmentation de n_estimators améliore le score et nécessite d'augmenter en combinaison le learning_rate (sensible)'
   → avec l'augmentation de n_estimators, le meilleur 'loss' peut changer
4. Au bout de quelques itérations, le meilleur score ne varie plus de manière sensible

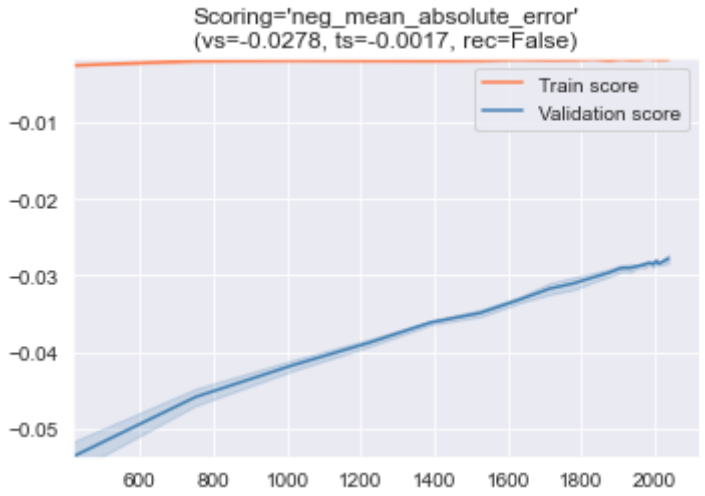


OPENCLASSROOMS
CentraleSupélec

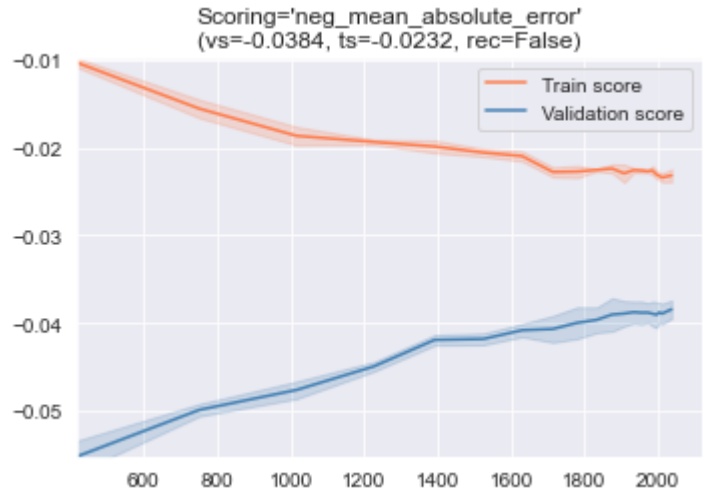
Améliorations effectuées (2/2)

Examen des courbes d'apprentissage en vue de la généralisation:

1. Le niveau d'erreur (MAE ou sqrt(MSE)) est faible
    1. Energie: 9265 kBtu, soit 0,12% de la moyenne de consommation en énergie des bâtiments
    2. Emissions : 0,3 tCO2e, soit 0,17% de la moyenne des émissions des bâtiments
2. Un écart important entre les courbes de training et de validation mais avec score croissant
3. Le gap entre les courbes pourrait être réduit:
    a. En diminuant la complexité du modèle (max_depth du DecisionTreeRegressor)
    b. En diminuant le nombre de features
    c. En augmentant la taille du jeu (sans EnergyStarScore), ce qui requerrait des mesures (données) complémentaires
4. Les solutions 3a et 3b n'apparaissent pas pertinentes compte tenu des points suivants:
    - L'échantillon de test a un bon score
    - La perte sur le score validation serait équivalente à la réduction du gap
    - La perte de features (catégories d'utilisation) potentiellement utiles pour générer des explications (ex: utilisation qui consomme le plus d'énergie)



Energie	Initial	Réduit	delta (i-r)
train score (MAE)	0.0017	0.0232	-0.0215
validation score (MAE)	0.0278	0.0384	-0.0106
delta(ts-vs)	0.0261	0.0152	0.0109



OPENCLASSROOMS
CentraleSupélec

Conclusion

En réponse à la problématique posée:

- Il est possible d'estimer l'énergie consommée et les émissions de GES avec un modèle à faible niveau d'erreur.

- L'estimation ne nécessite pas de disposer de l'"ENERGY STAR Score".

- La généralisation nécessitera un pré-traitement des données, s'agissant en particulier des surfaces associées aux principales utilisations, afin d'assurer leur cohérence avec les surfaces totales.

Echanges avec l’évaluateur

- **Discussion**
- **Débriefing**

OPENCLASSROOMS CentraleSupélec

Contact: Eric TREGOAT

eric.tregoat@gmail.com

06 49 99 79 59

https://www.linkedin.com/in/erictregoat/