Projet 4 : Anticipez les besoins en consommation électrique de bâtiments

Mentors : Morgan MOISON et Amine HADJ-YOUCEF

Etudiant : Marin DUCHEMIN

OPENCLASSROOMS

Plan de la Présentation

I ~ Présentation de la problématique

II ~ Préparation des données

III ~ Modélisations testées

IV ~ Modèle final choisi

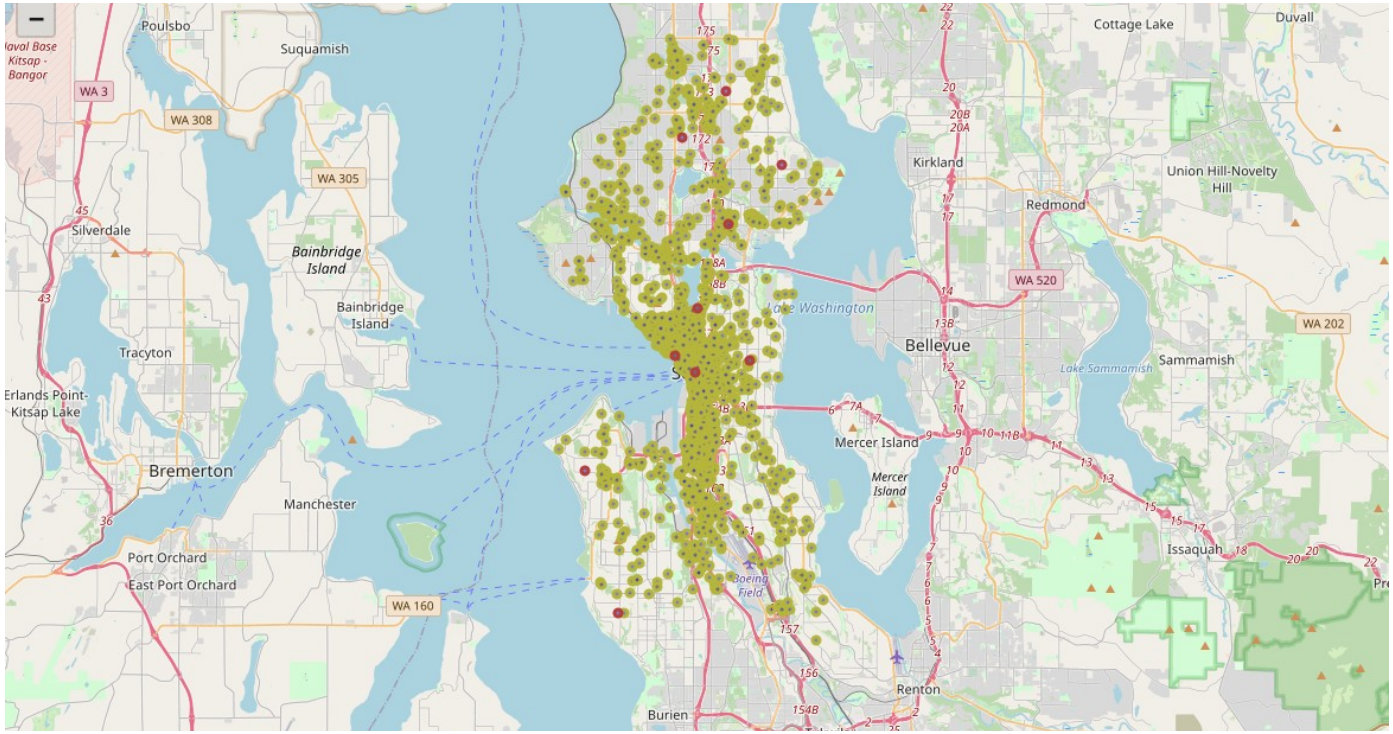
I ~ Présentation de la problématique

Seattle : Ville neutre en émission carbone

Objectif : Prédiction des émissions de gas à effets de serre et de la consommation totale d’énergie de bâtiment

I ~ Présentation de la problématique

Carte des bâtiments avec relevés (vert moutarde) et sans relevés (rouge)



I ~ Présentation de la problématique

- Pour les bâtiments non destinés à l’habitation
- Grâce aux données déclaratives du permis d’exploitation commerciale
- Recherche de l’intérêt de l’ENERGYSTARScore, indicateur complexe à obtenir

II ~ Préparation des données

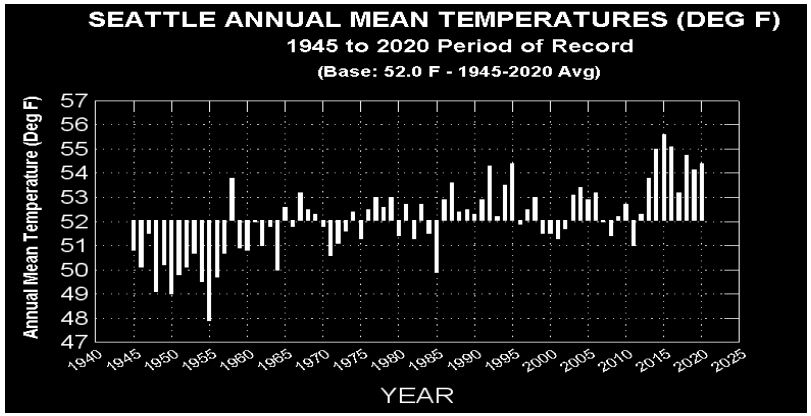
- Suppression des données non utilisées dans le projet (habitation, données non déclaratives, etc)
- Passage aux unités du SI
- Suppression de colonnes jugées inutiles

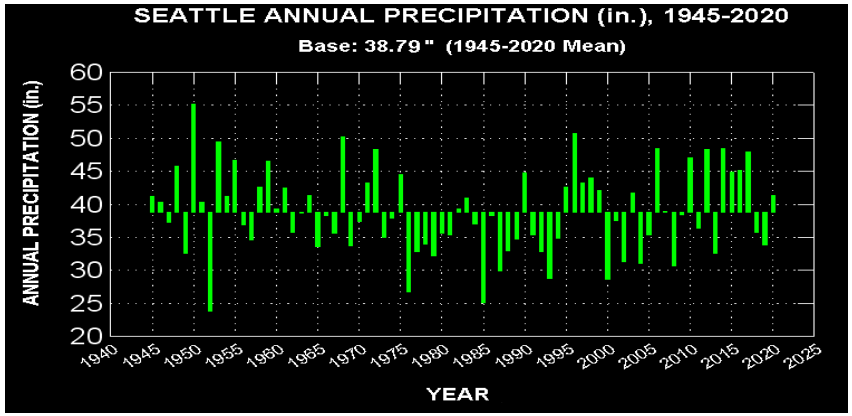
```
RangeIndex: 3340 entries, 0 to 3339
Data columns (total 47 columns):
 #   Column                                                  Non-Null Count  Dtype
---  ------                                                  --------------  -----
 0   OSEBuildingID                                           3340 non-null   int64
 1   DataYear                                                3340 non-null   int64
 2   BuildingType                                            3340 non-null   object
 3   PrimaryPropertyType                                     3340 non-null   object
 4   PropertyName                                            3340 non-null   object
 5   TaxParcelIdentificationNumber                           3338 non-null   object
 6   Location                                                3340 non-null   object
 7   CouncilDistrictCode                                     3340 non-null   int64
 8   Neighborhood                                            3340 non-null   object
 9   YearBuilt                                               3340 non-null   int64
 10  NumberofBuildings                                       3340 non-null   int64
 11  NumberofFloors                                          3332 non-null   float64
 12  PropertyGFATotal                                        3340 non-null   int64
 13  PropertyGFAParking                                      3340 non-null   int64
 14  PropertyGFABuilding(s)                                  3340 non-null   int64
 15  ListOfAllPropertyUseTypes                               3213 non-null   object
 16  LargestPropertyUseType                                  3204 non-null   object
 17  LargestPropertyUseTypeGFA                               3204 non-null   float64
 18  SecondLargestPropertyUseType                            1559 non-null   object
 19  SecondLargestPropertyUseTypeGFA                         1559 non-null   float64
 20  ThirdLargestPropertyUseType                             560 non-null    object
 21  ThirdLargestPropertyUseTypeGFA                          560 non-null    float64
 22  YearsENERGYSTARCertified                                110 non-null    object
 23  ENERGYSTARScore                                         2560 non-null   float64
 24  SiteEUI(kBtu/sf)                                        3330 non-null   float64
 25  SiteEUIWN(kBtu/sf)                                      3330 non-null   float64
 26  SourceEUI(kBtu/sf)                                      3330 non-null   float64
 27  SourceEUIWN(kBtu/sf)                                    3330 non-null   float64
 28  SiteEnergyUse(kBtu)                                     3330 non-null   float64
 29  SiteEnergyUseWN(kBtu)                                   3330 non-null   float64
 30  SteamUse(kBtu)                                          3330 non-null   float64
 31  Electricity(kWh)                                        3330 non-null   float64
 32  Electricity(kBtu)                                       3330 non-null   float64
 33  NaturalGas(therms)                                      3330 non-null   float64
 34  NaturalGas(kBtu)                                        3330 non-null   float64
 35  OtherFuelUse(kBtu)                                      3330 non-null   float64
 36  GHGEmissions(MetricTonsCO2e)                            3330 non-null   float64
 37  GHGEmissionsIntensity(kgCO2e/ft2)                       3330 non-null   float64
 38  DefaultData                                             3339 non-null   object
 39  Comment                                                 13 non-null     object
 40  ComplianceStatus                                        3340 non-null   object
 41  Outlier                                                 84 non-null     object
 42  2010 Census Tracts                                      224 non-null    float64
 43  Seattle Police Department Micro Community Policing Plan Areas  3338 non-null   float64
 44  City Council Districts                                  213 non-null    float64
 45  SPD Beats                                               3338 non-null   float64
 46  Zip Codes                                               3340 non-null   int64
dtypes: float64(23), int64(9), object(15)
memory usage: 1.2+ MB
```

II ~ Préparation des données

Fusion des deux jeux de données.

2015 et 2016, années similaires pour Seattle.

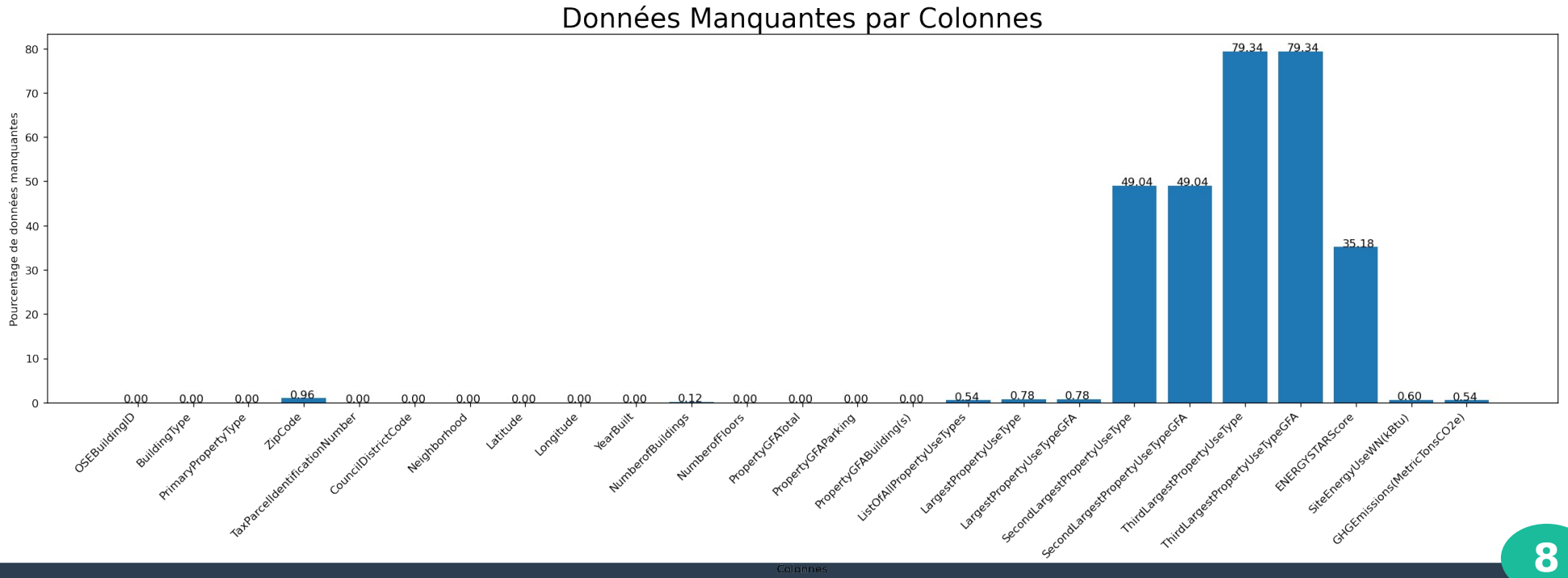




2016 pour les données qualitatives et moyenne des deux années pour les données numériques.

II ~ Préparation des données

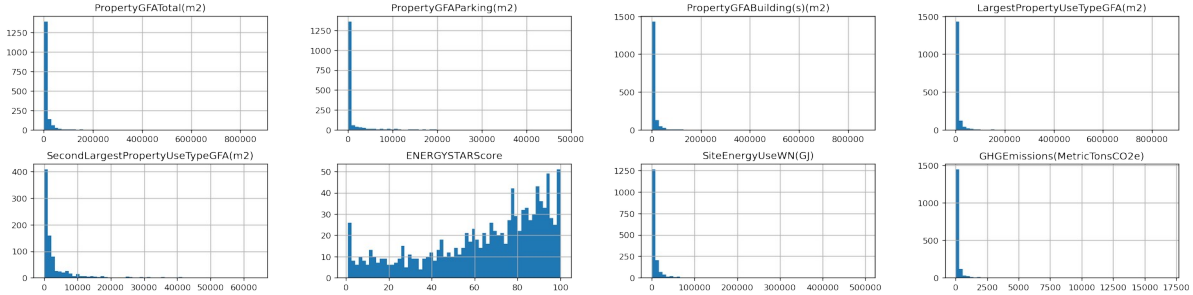
Suppression des colonnes avec trop de données manquantes.



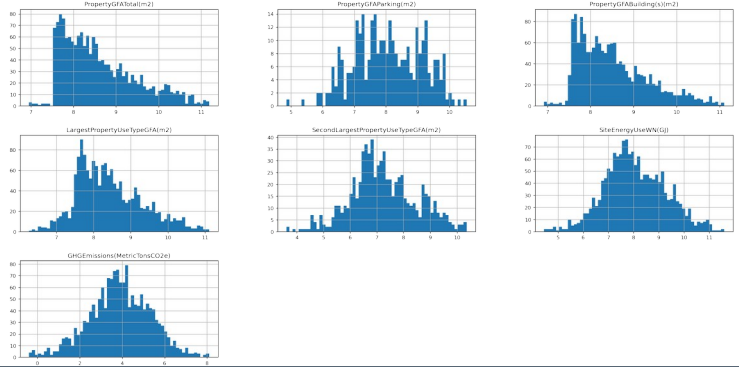
II ~ Préparation des données

Normalisation des distributions puis suppression des cas particuliers.

- Avant :

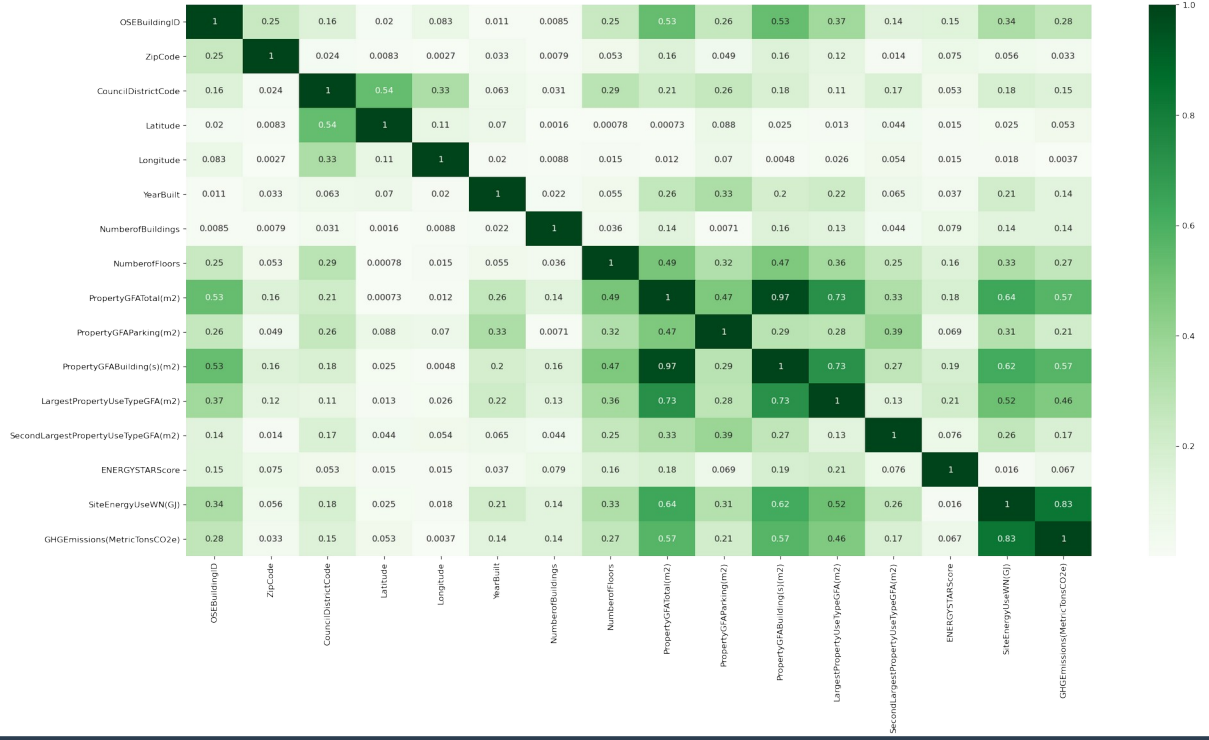


- Après :



II ~ Préparation des données

Recherche de corrélations entre les variables.



III ~ Modélisations testées

Trois modèles testés :

- Kernel Ridge Regression
- Random Forest
- Gradient Boosting

III ~ Modélisations testées

Pour chaque modèle, on teste avec plusieurs entrainements différents :

- Vanilla (variables de l’exploration sauf l’EnergyStarScore)
- Avec ENERGYSTARSCORE (variables dont l’EnergyStarScore)
- Moins de Variables (variables les plus importantes)
- Hyperparamètres optimisés (via une validation croisée sur le Vanilla)

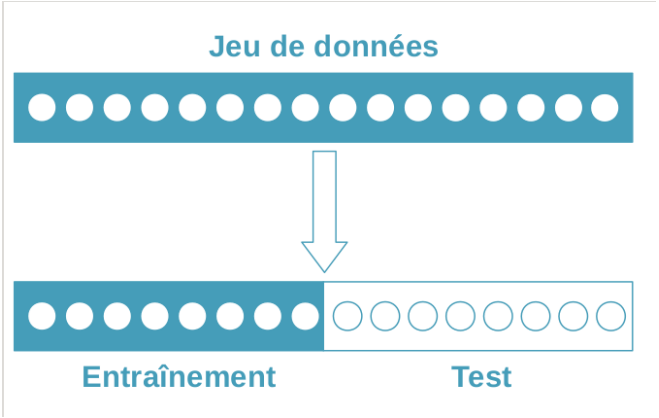
III ~ Modélisations testées

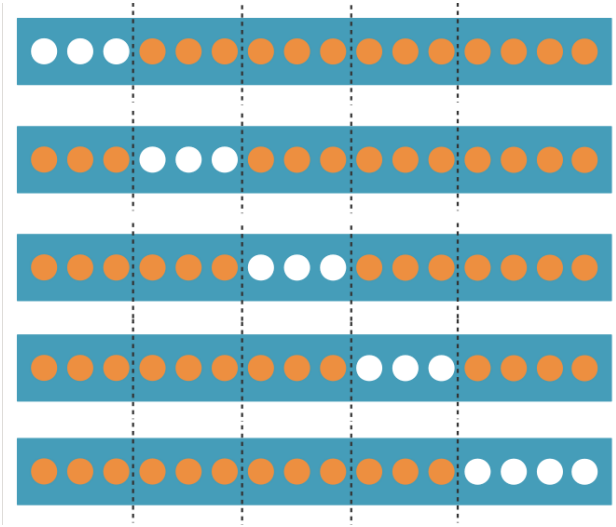
Les données subissent préalablement un pré-traitement :

- Toutes les données sont retravaillées avec SimpleImputer de “scikit-learn”.
- Les données catégorielles sont encodées via OneHotEncoder de “scikit-learn”.
- Les données numériques sont mises à l’échelle via RobustScaler de “scikit-learn”.

III ~ Modélisations testées

Pour chaque modèle, on sépare les données en jeu d'entrainement et en jeu de test.



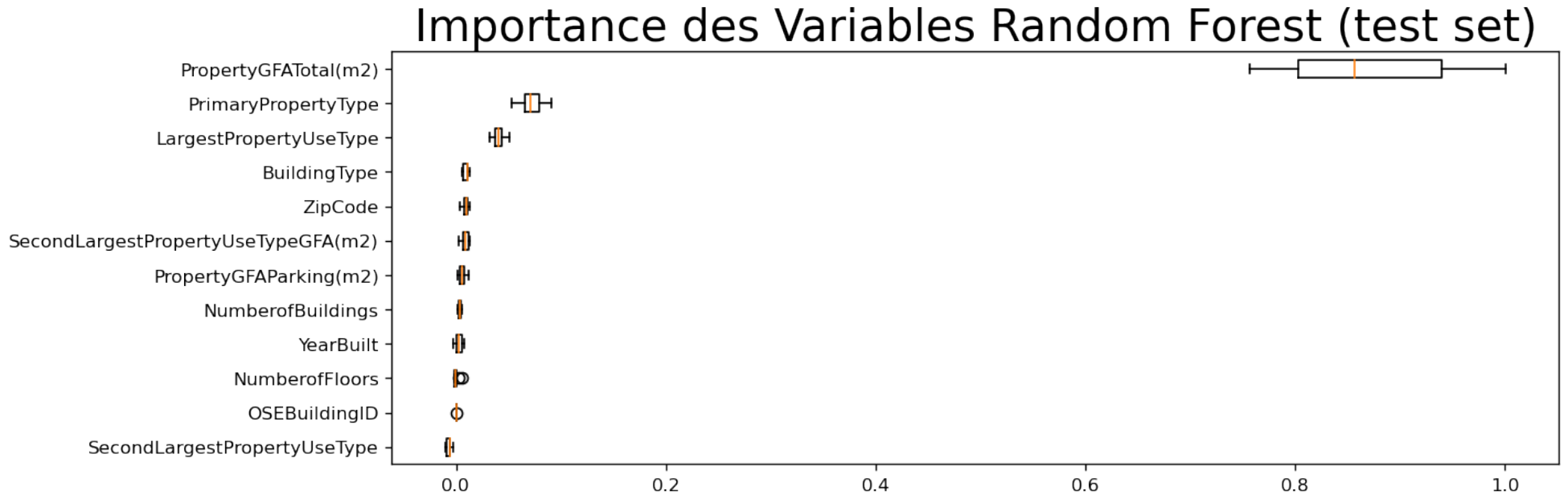


Pour chaque recherche des hyperparamètres, on fait une validation croisée.

Images prises sur le cours d'OpenClassRooms "Evaluez les performances d'un modèle de machine learning".

III ~ Modélisations testées

Sélection des variables les plus importantes (exemple avec le modèle “Random Forest”) :



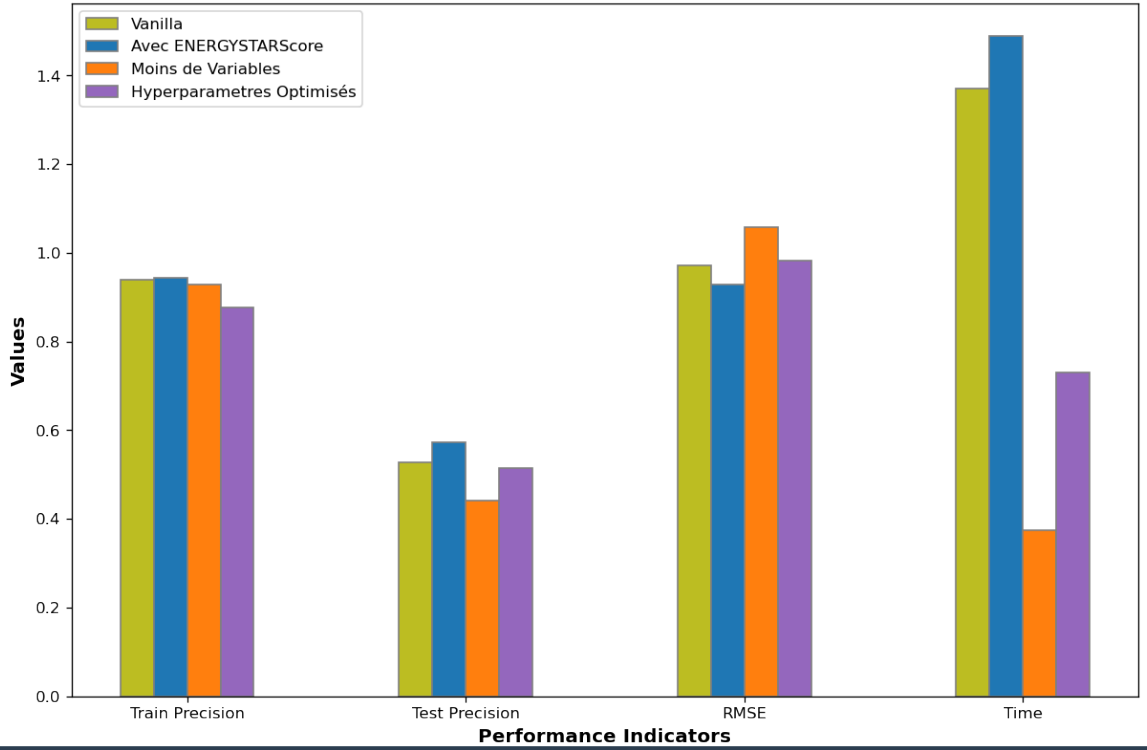
III ~ Modélisations testées

Kernel Ridge Regression :

- Regression non linéaire
- Minimisation de l’erreur quadratique entre la prédiction et la réalité
- Un peu plus lente qu’une Support Vector Regression (SVR)

III ~ Modélisations testées

Résultats



Trained Data	Train Precision	Test Precision	RMSE	Time
Vanilla	0.924380	0.513767	0.981673	0.112739
Avec ENERGYSTARScore	0.932660	0.568779	0.927708	0.119539
Moins de Variables	0.630628	0.571942	0.924087	0.108947
Hyperparametres Optimisés	0.949026	0.514063	0.981908	0.169959

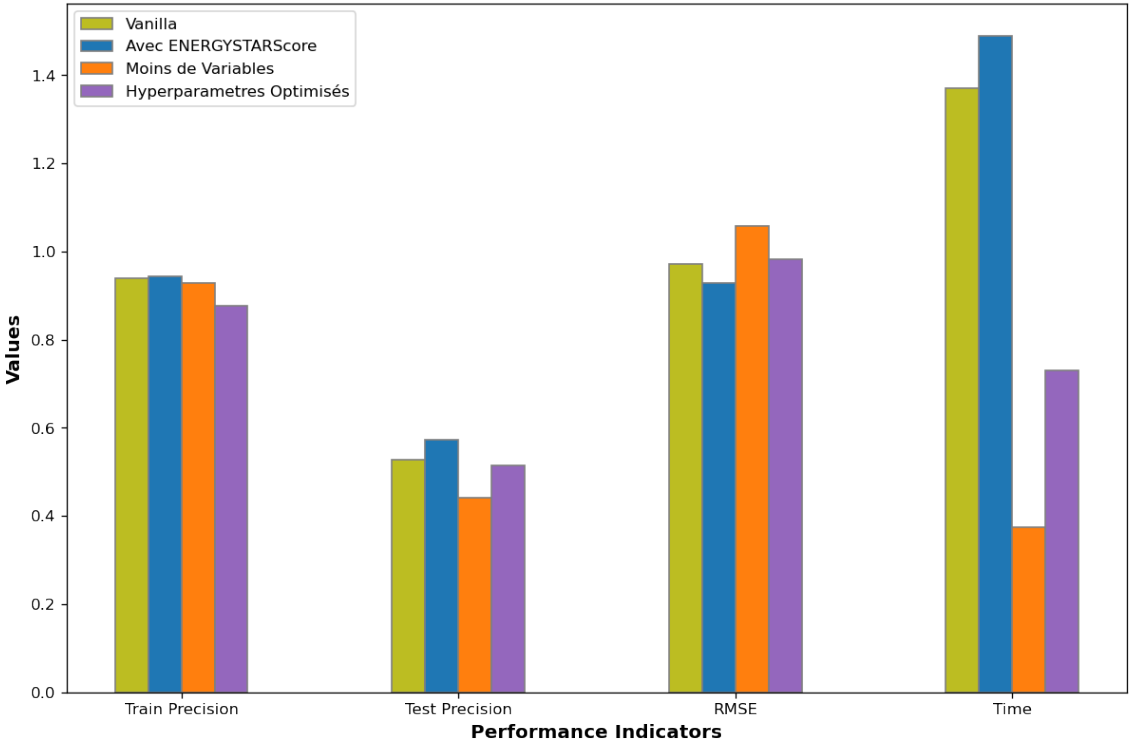
III ~ Modélisations testées

Random Forest :

- Méthode ensembliste
- Arbres de décision en parallèle
- Méthode très utilisée car assez générale

III ~ Modélisations testées

Résultats



Trained Data	Train Precision	Test Precision	RMSE	Time
Vanilla	0.938365	0.527356	0.971168	1.370676
Avec ENERGYSTARScore	0.944224	0.572100	0.927605	1.488481
Moins de Variables	0.927621	0.440726	1.057483	0.373760
Hyperparametres Optimisés	0.877172	0.515448	0.983454	0.729277

III ~ Modélisations testées

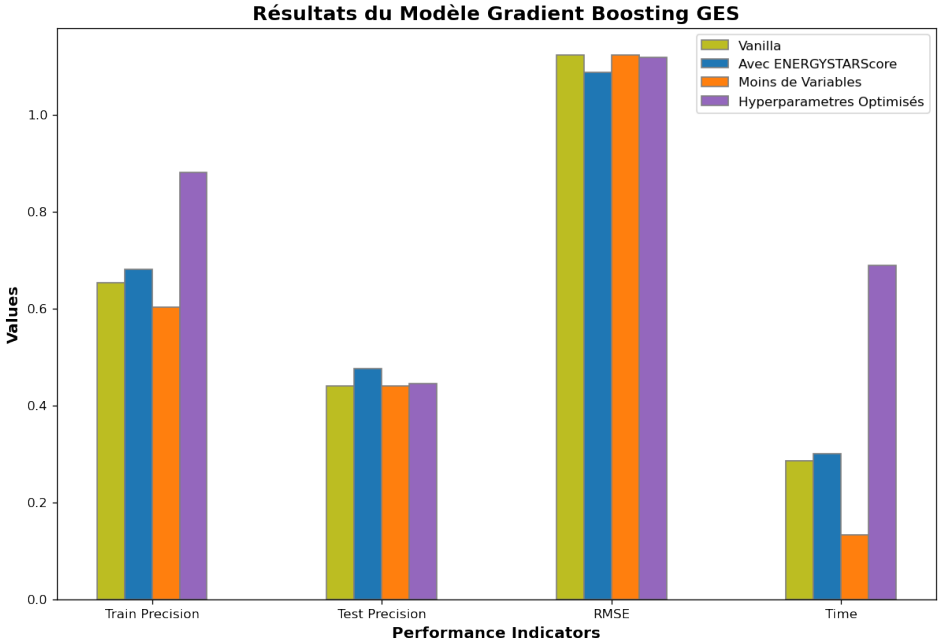
Gradient Boosting :

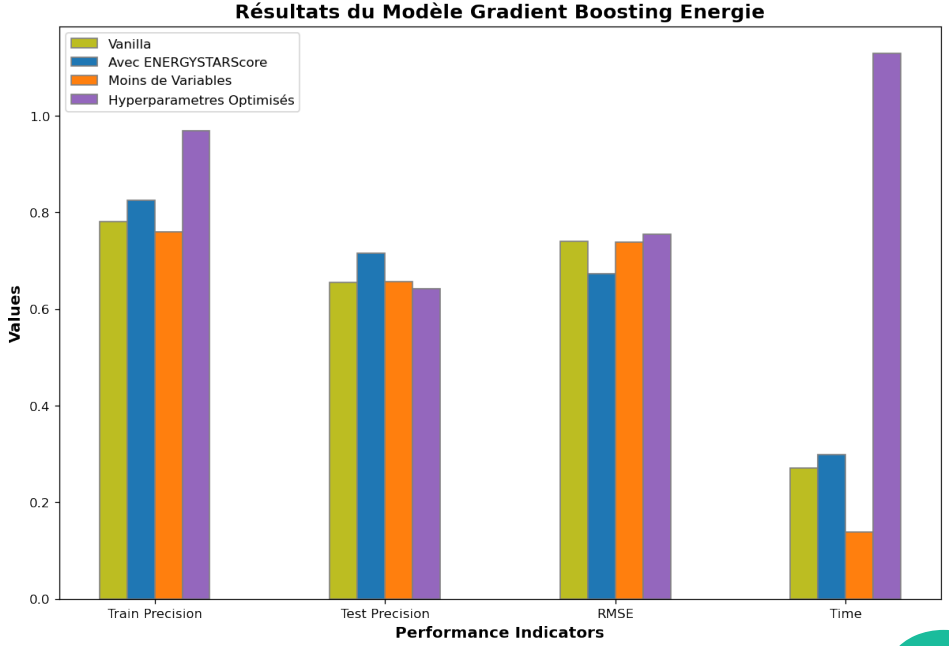
- Méthode ensembliste
- Arbres de décision en séquentiel
- Minimisation de la fonction de perte (des moindres carrés)

III ~ Modélisations testées

Résultats

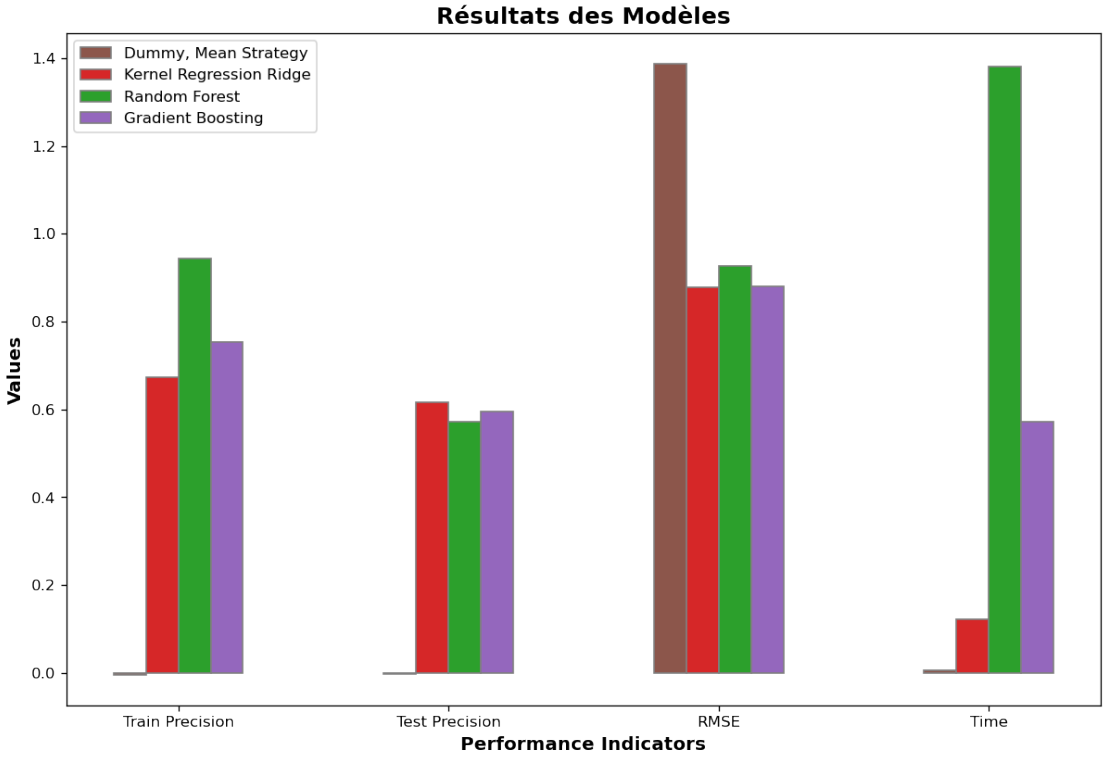
Gradient Boosting GES					Gradient Boosting Energie				
Trained Data	Train Precision	Test Precision	RMSE	Time	Trained Data	Train Precision	Test Precision	RMSE	Time
Vanilla	0.653075	0.440656	1.122232	0.285365	Vanilla	0.780839	0.654786	0.740803	0.270540
Avec ENERGYSTARScore	0.680182	0.475491	1.086724	0.300622	Avec ENERGYSTARScore	0.825197	0.715071	0.673019	0.298754
Moins de Variables	0.603102	0.440606	1.122281	0.132389	Moins de Variables	0.759459	0.656561	0.738897	0.137699
Hyperparametres Optimisés	0.881054	0.445177	1.117687	0.688642	Hyperparametres Optimisés	0.969210	0.642245	0.754139	1.128745





III ~ Modèle final choisi

Conclusion



Model	Train Precision	Test Precision	RMSE	Time
Dummy, Median Strategy	-0.003830	-0.001997	1.387284	0.006019
Kernel Regression Ridge	0.672816	0.616714	0.877935	0.122230
Random Forest	0.944224	0.572100	0.927605	1.381324
Gradient Boosting	0.752689	0.595281	0.879871	0.572690
Gradient Boosting GES	0.680182	0.475491	1.086724	0.294220
Gradient Boosting Energie	0.825197	0.715071	0.673019	0.278470

III ~ Modélisations testées

Conclusion

- Kernel Ridge Regression choisi pour ses performances

- EnergyStarScore permet un gain de précision d’environ 10%

OSEBuildingID	SiteEnergyUseWN(GJ)	TotalGHGEmissions(MetricTonsCO2e)
87.0	2721.323874	59.932394
757.0	6253.905264	105.070765
773.0	2973.490743	41.603089
19694.0	2408.796258	56.450390
20130.0	811.136674	17.598811
20324.0	1610.651482	37.827599
21570.0	2408.796258	56.450390
21578.0	2721.323874	59.932394
24408.0	6269.054177	142.171272
25361.0	2408.796258	56.450390
25451.0	4013.165248	111.668142
26532.0	2973.490743	55.441249
49968.0	3359.284257	56.450390
49972.0	12616.984128	226.370240

III ~ Modélisations testées

Axes d’améliorations :

- Pousser les modèles ensemblistes pour comprendre leurs relatives faibles performances (surtout les hyperparamètres)
- Utiliser une autre méthode pour sélectionner les variables d’intérêt (automatisation via machine learning?)
- Tester d’autres modèles supervisés (réseau de neurones) ou non.

Merci de votre attention