Seattle :Consommation et émissions des bâtiments NON destinés à l'habitation.

Projet 3

Sofia Velasco

Analyse exploratoire

A. Caractéristiques générales du jeu de données et traitement initial

- **2 années (2015 et 2016)→ 2 dataset :**

	2015	2016
Colonnes (Variables)	47	46
Lignes	3340	3376

- **Homogénéisation des colonnes** → Colonnes différentes entre les 2 années **:**

  - Split de la colonne « Location » (2015)
  - Coïncidence des noms des colonnes
  - Mise en ordre alphabétique
  - Coïncidence des types associés à chaque colonnes

- **Regroupement des données dans un seul dataset** → chaque bâtiment de chaque année est une entité nouvelle et indépendante:

  46 Colonnes, 6716 Lignes.

B. Nettoyage des données

- **Comprendre les variables:**

  - suffixes WN : "Weather Normalized" → la météo n'est pas à considérer ✗
  - GHG → CO2 ✓
  - GFA → Surface totale au sol ✓
  - EUI → Energie consommée par pied carré par an (kBtu/sf → Kilo-British thermal unit) ✓
  - Site Energy → Chaleur et énergie consommée présente dans la facture ✗
  - Source Energy → Quantité totale de combustible brut nécessaire au fonctionnement du bâtiment (transport depuis la source) ✓

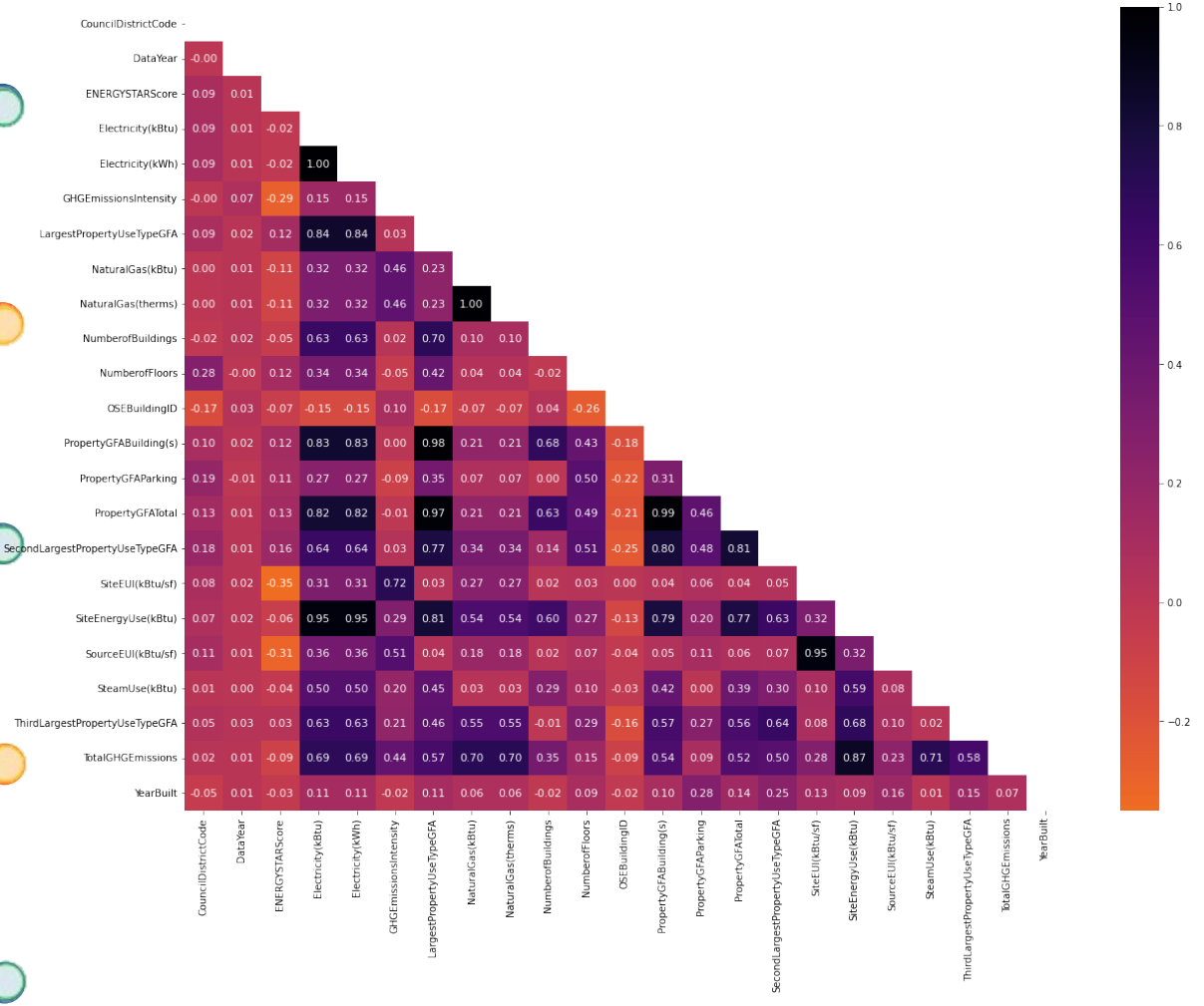
- **Élimination des variables avec les suffixes 'WN'**
- **Élimination des lignes 'Outliers'** → on ignore à quoi ils font référence

43 variables:
23 quantitatives,
20 qualitatives

- **Élimination des 'BuildingType' "NON DESTINES A L'HABITATION"** → suffixes 'Multifamily'.
- **Mise en format Python des noms des variables** ('SourceEUI(kBtu/sf)' → 'SourceEUI_kBtu_sf')

C. Analyse des données

C.1 Corrélations entre les 43 variables → Pearson pour les quantitatives ANOVA pour les qualitatives

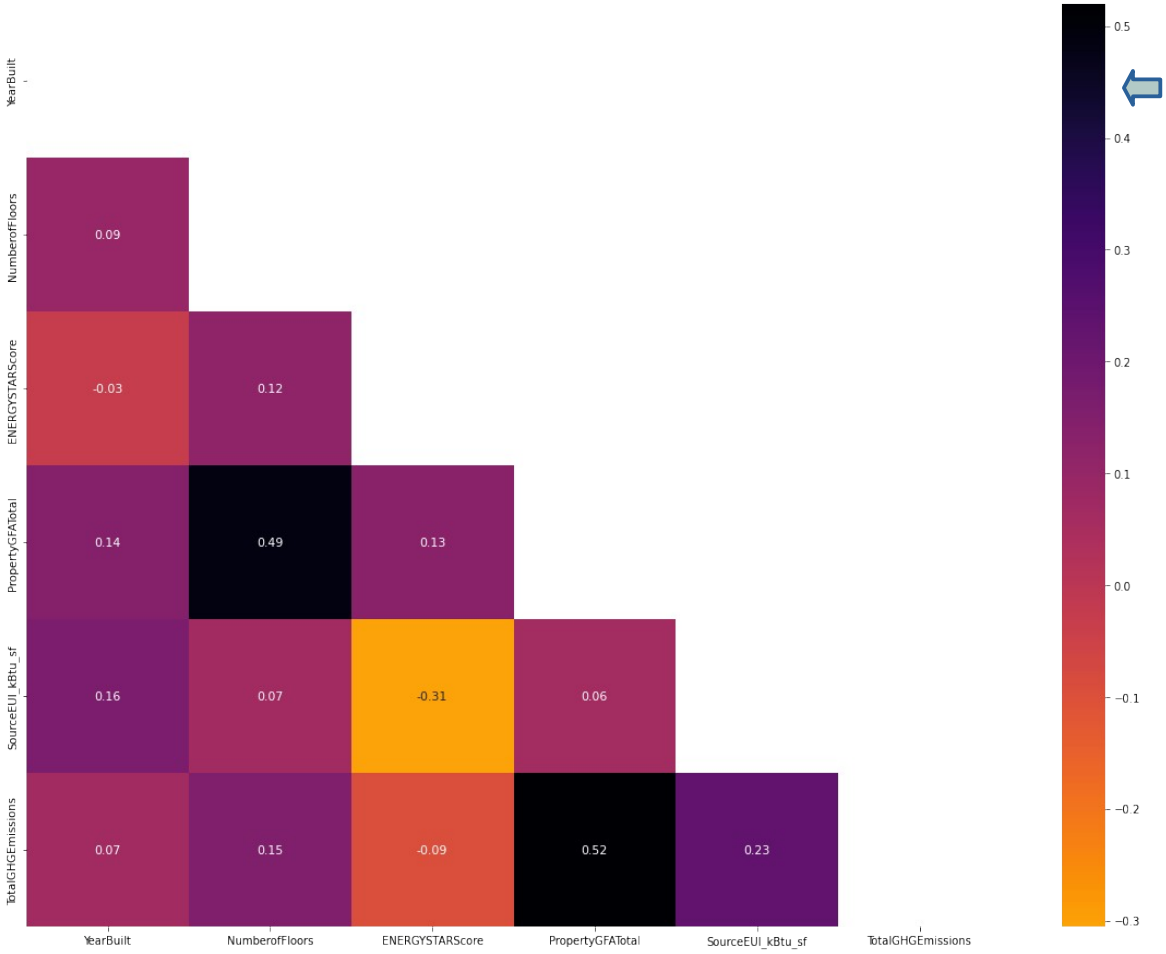


Pearson pour les 23 quantitatives: corrélation importante si sa valeur absolue est >= 0.6

Variables intéressantes :
YearBuilt,
NumberofFloors,
ENERGYSTARScore,
PropertyGFATotal,
SourceEUI(kBtu/sf),
TotalGHGEmissions

C. Analyse des données

Corrélations entre les 6 variables quantitatives intéressantes



Pas de grandes corrélations entre nos variables, juste 'PropertyGFA Total' qui est légèrement corrélée avec 'TotalGHGEmissions' et avec 'NumberofFloors'

Parmi les variables qualitative 3 intéressantes:
'BuildingType',
'Neighborhood'
'ZipCode'

ANOVA pour les 3 qualitatives:
Leur ANOVA avec chacune des 6 variables quantitatives sélectionnées montre existence de corrélation (ie. P<<0.05). Logique mais pas grave: ce qui nous intéresse c'est si ces variables décrivent la tendance en dépense énergétique et CO2 (ie. leur corrélation avec SourceEUI_kBtu_sf et TotalGHGEmissions).

Variables Sélectionnées :

0. Variable pour différentier les deux data sets: DataYear

1. Variables d'identification des bâtiments: OSEBuildingID, TaxParcelIdentificationNumber, PropertyName

2. Variables potentiellement explicatives du caractère éco-responsable:

2.1 propre aux bâtiments:

*BuildingType, °YearBuilt, °NumberofFloors, °PropertyGFATotal et °ENERGYSTARScore → (on veut la tester)

2.2 propre aux zones où se trouvent les bâtiments:

*Neighborhood, *ZipCode

3. Variables cibles (2 variables qui mesurent le caractère éco-responsable des bâtiments): °SourceEUI(kBtu/sf) (dépense totale d'énergie), °TotalGHGEmissions (CO2)

'°' → variables quantitatives
'*' → variables qualitatives

B. Nettoyage des données

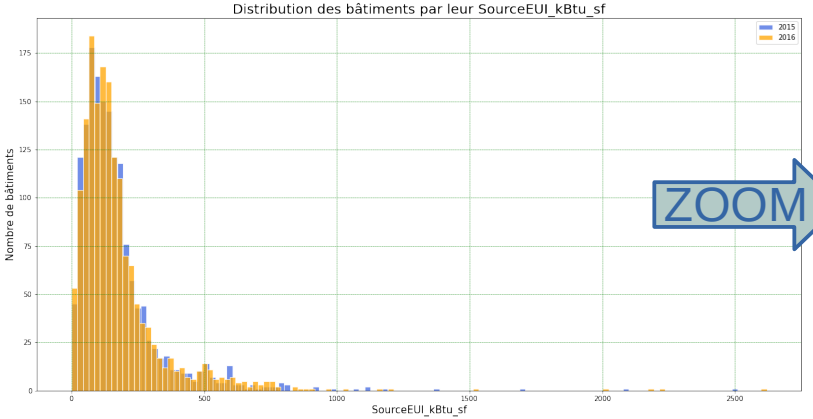
Identification et élimination des données manquantes

- 'SourceEUI(kBtu/sf)' et 'TotalGHGEmissions' → **moins de 0.3% de NaN ont les élimine car il s'agit de nos variables cibles.**
- 'TaxParcelIdentificationNumber' et 'NumberofFloors' → **moins de 0.3% de NaN.** A la fin de tous les traitements si il en reste on les élimine (on ne peut pas les inventer)
- **'ENERGYSTARScore' → près de 33.5% de NaN, première faiblesse de cette variable.** Lorsque on prendra en compte cette variable il faudra les éliminer car on ne peut pas les remplacer.

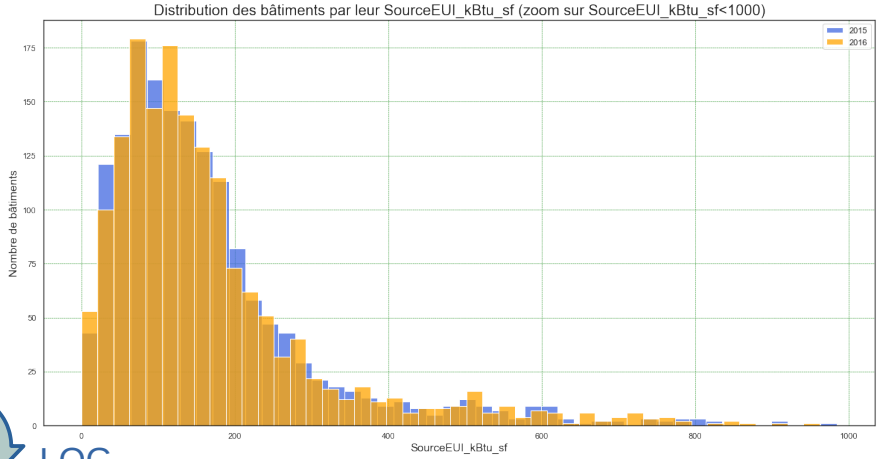
Variables Sélectionnées	Nombre de NaN
DataYear	0
OSEBuildingID	0
TaxParcelIdentificationNumber	1
PropertyName	0
BuildingType	0
YearBuilt	0
NumberofFloors	8
PropertyGFATotal	0
ENERGYSTARScore	1095
SourceEUI_kBtu_sf	**7**
TotalGHGEmissions	**7**
Neighborhood	0
ZipCode_New	0

D. Distributions des variables cibles

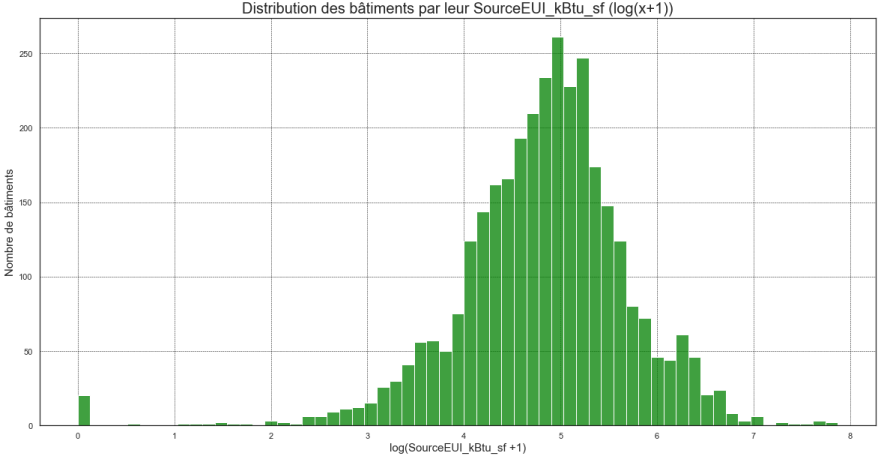
SourceEUI_kBtu_sf



ZOOM



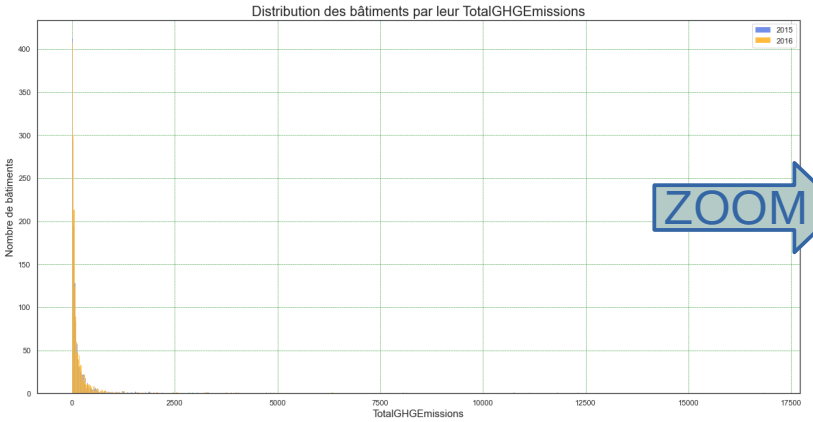
LOG



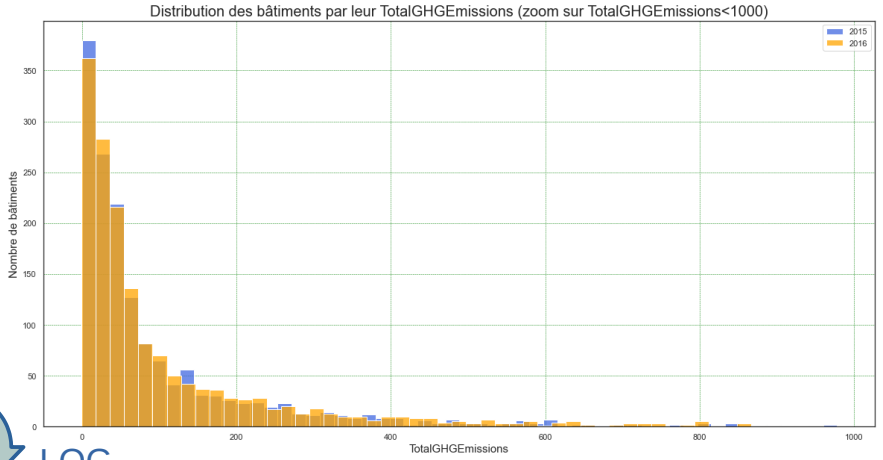
Afin de rapprocher des valeurs extrêmes pour obtenir un graphique de distribution moins étendu on fait log(x+1)

D. Distributions des variables cibles

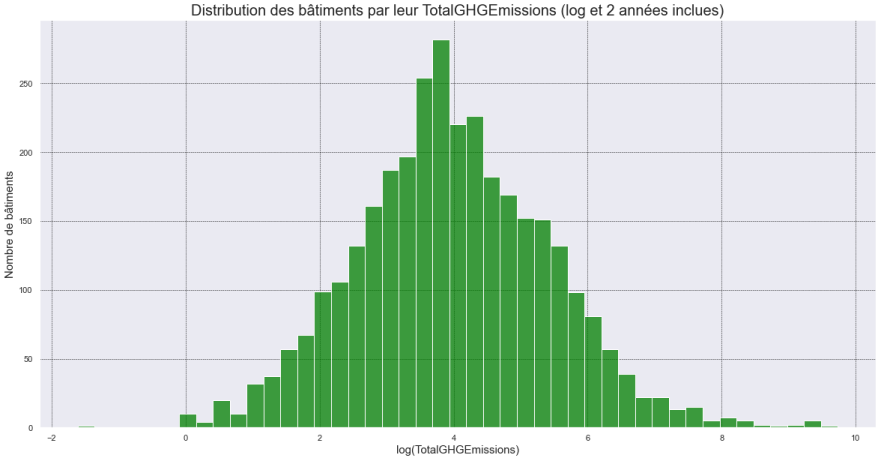
TotalGHGEmissions



ZOOM



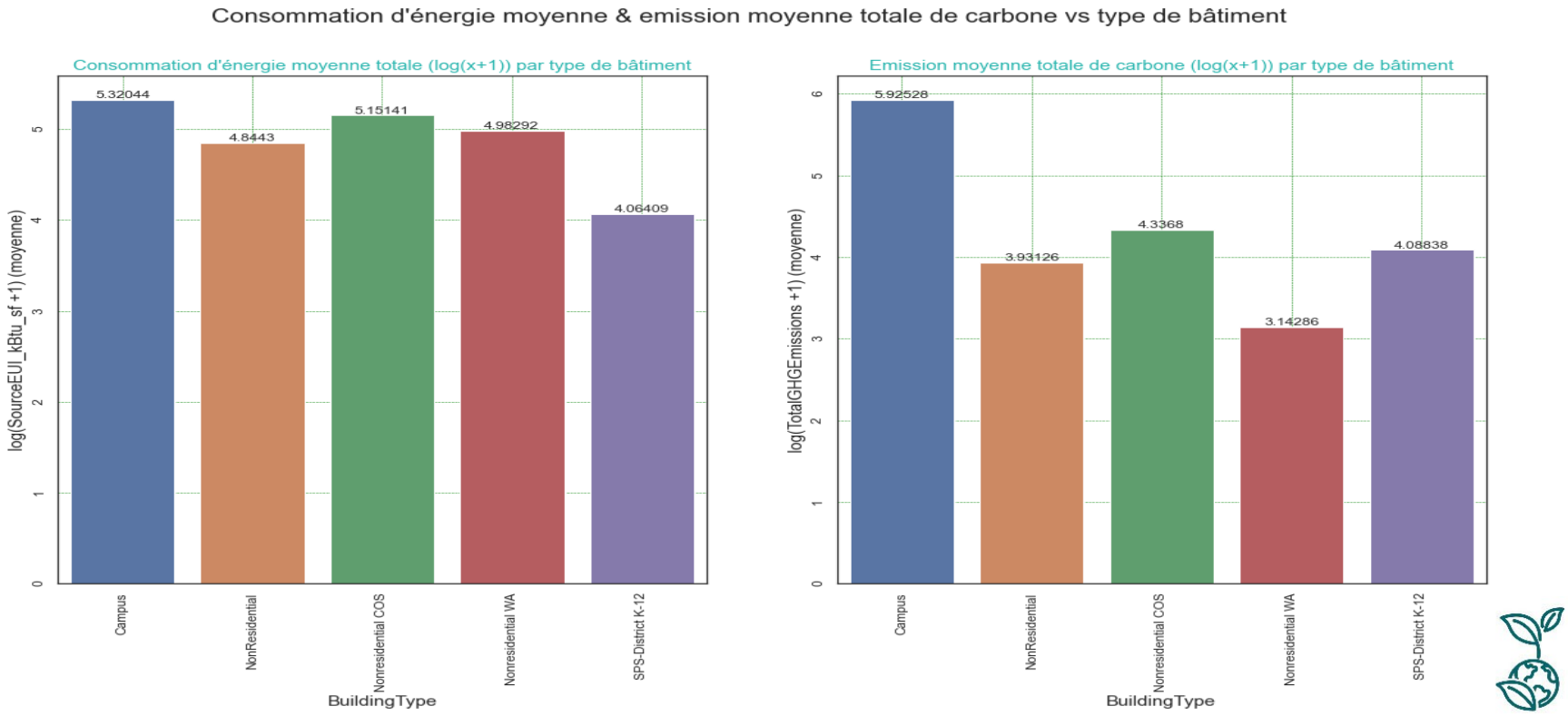
LOG



Afin de rapprocher des valeurs extrêmes pour obtenir un graphique de distribution moins étendu on fait log(x+1)

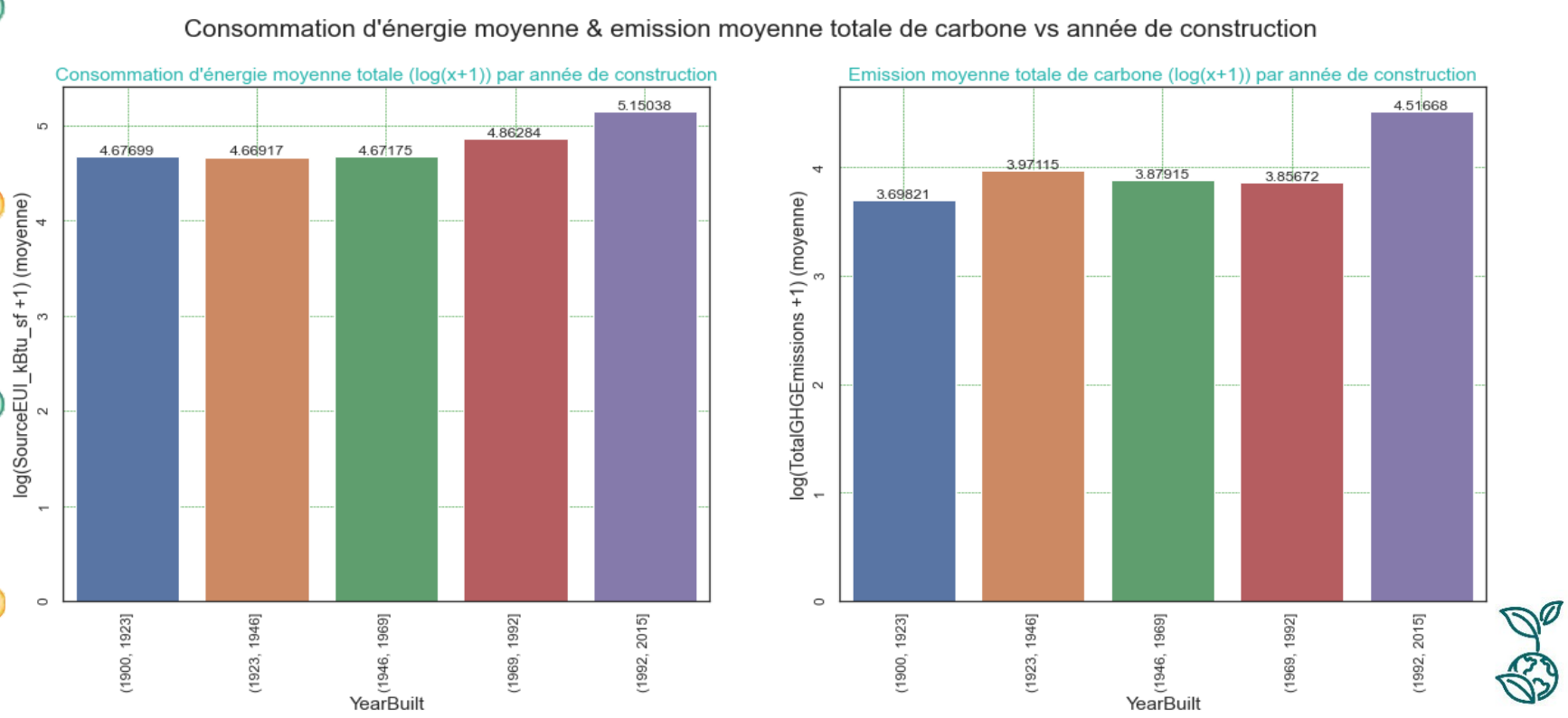
D. Distributions des variables explicatives (features)

BuildingType: Les 'Campus' ressortent sont les plus et consommateurs d'énergie et émetteurs de carbone.



D. Distributions des variables explicatives (features)

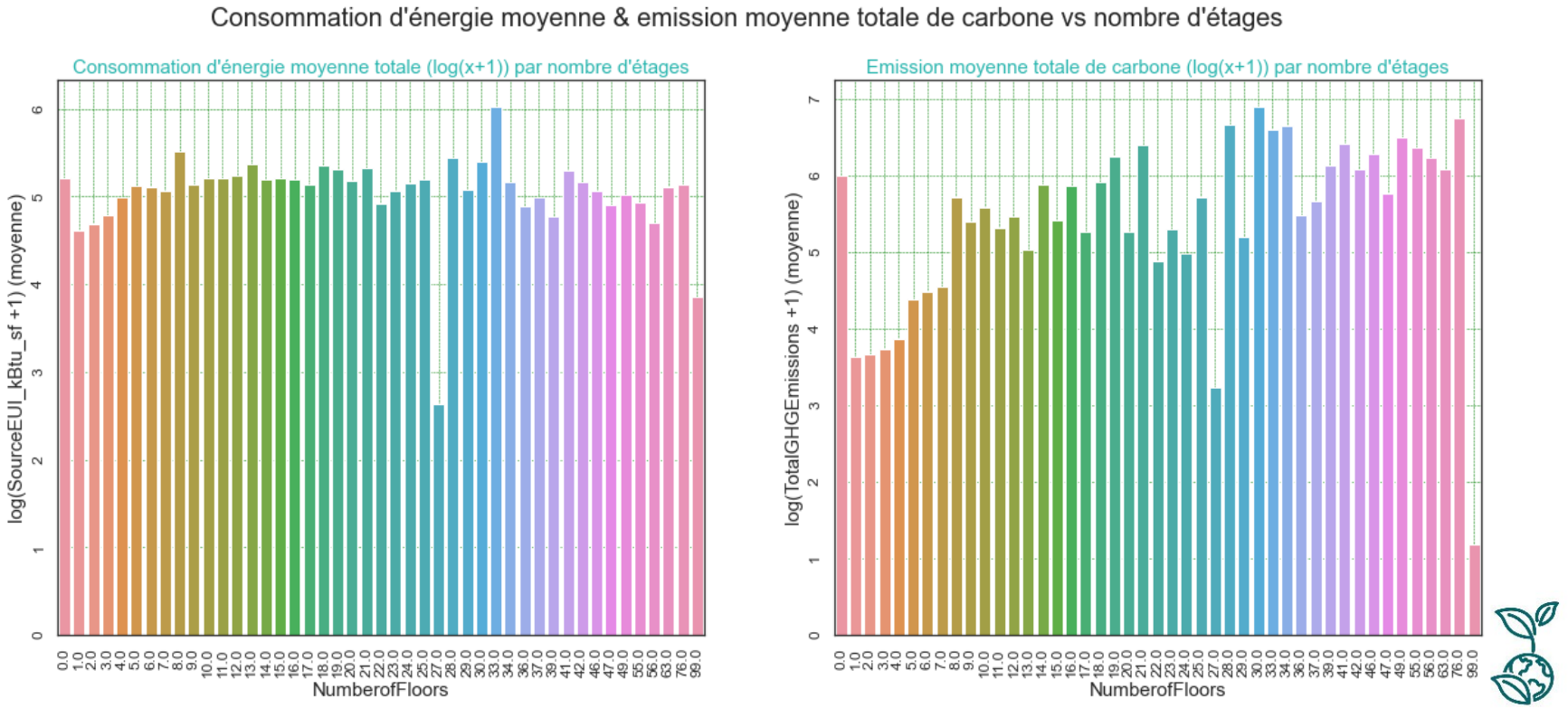
YearBuilt: les bâtiments les bâtiments de moins de 23 ans (construits entre 1992 et 2015) consomment le plus d'énergie et émettent le plus de carbone.



On fait des « bins » (tranches): ((max-min)/5)

D. Distribution des variables explicatives (features)

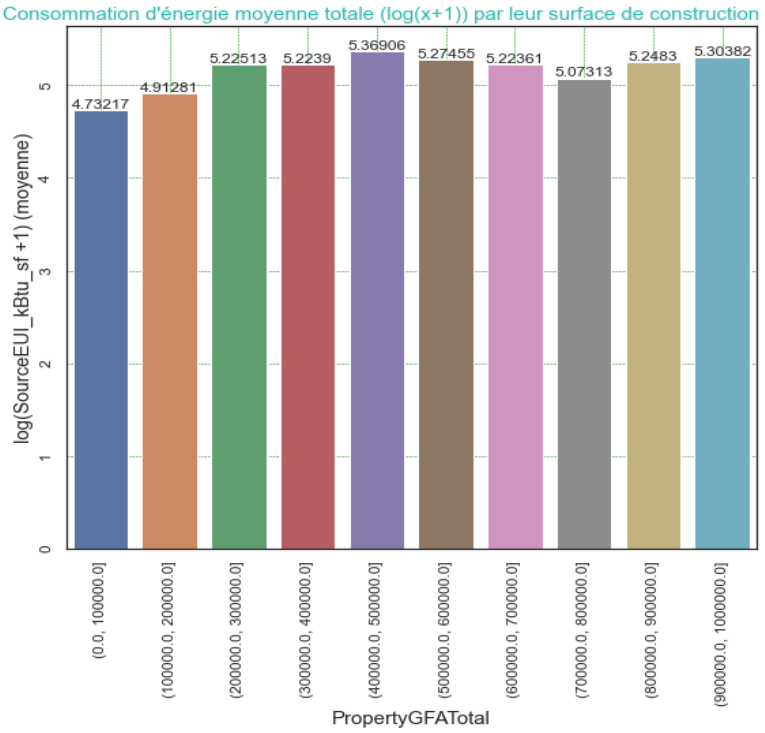
NumberofFloors: 99 étages → moins de consommation d'énergie et d'émission de CO2? → pas de sens → variable à ne pas considérer.

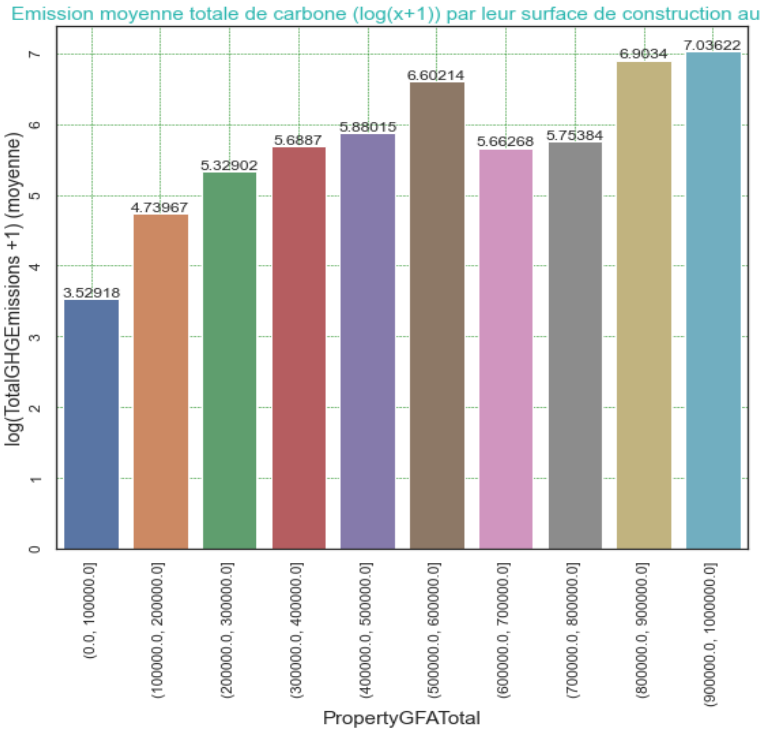


D. Distributions des variables explicatives (features)

PropertyGFATotal: les bâtiments entre 800 000 et 1 000 000 m^2 parmi ceux qui consomment le plus d'énergie et ceux qui émettent le plus de carbone. Jusqu'à 200 000 m^2 c'est le contraire.

Consommation d'énergie moyenne & emission moyenne totale de carbone vs surface de construction au sol

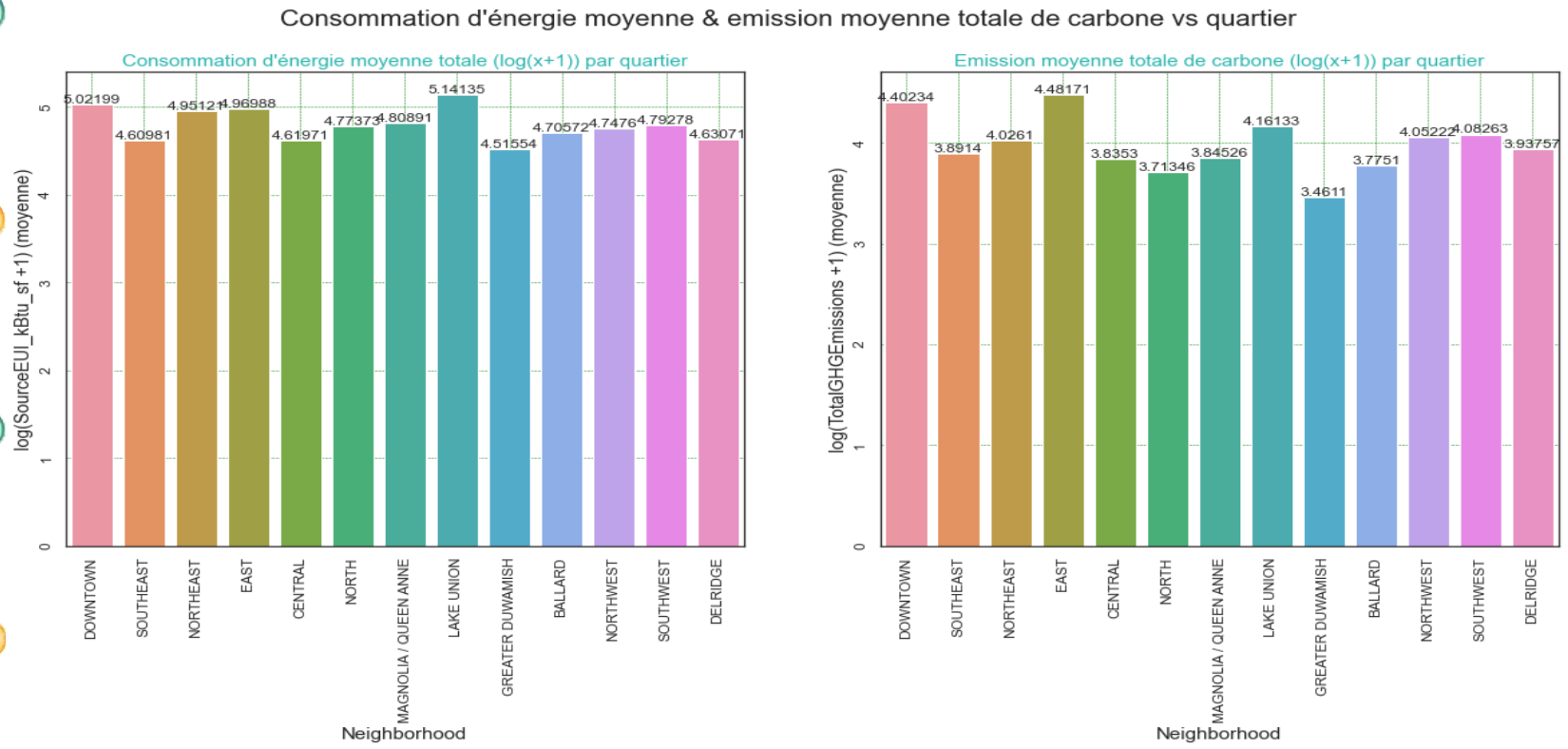




On fait des « bins » (tranches): chaque 0.1e6 m^2:

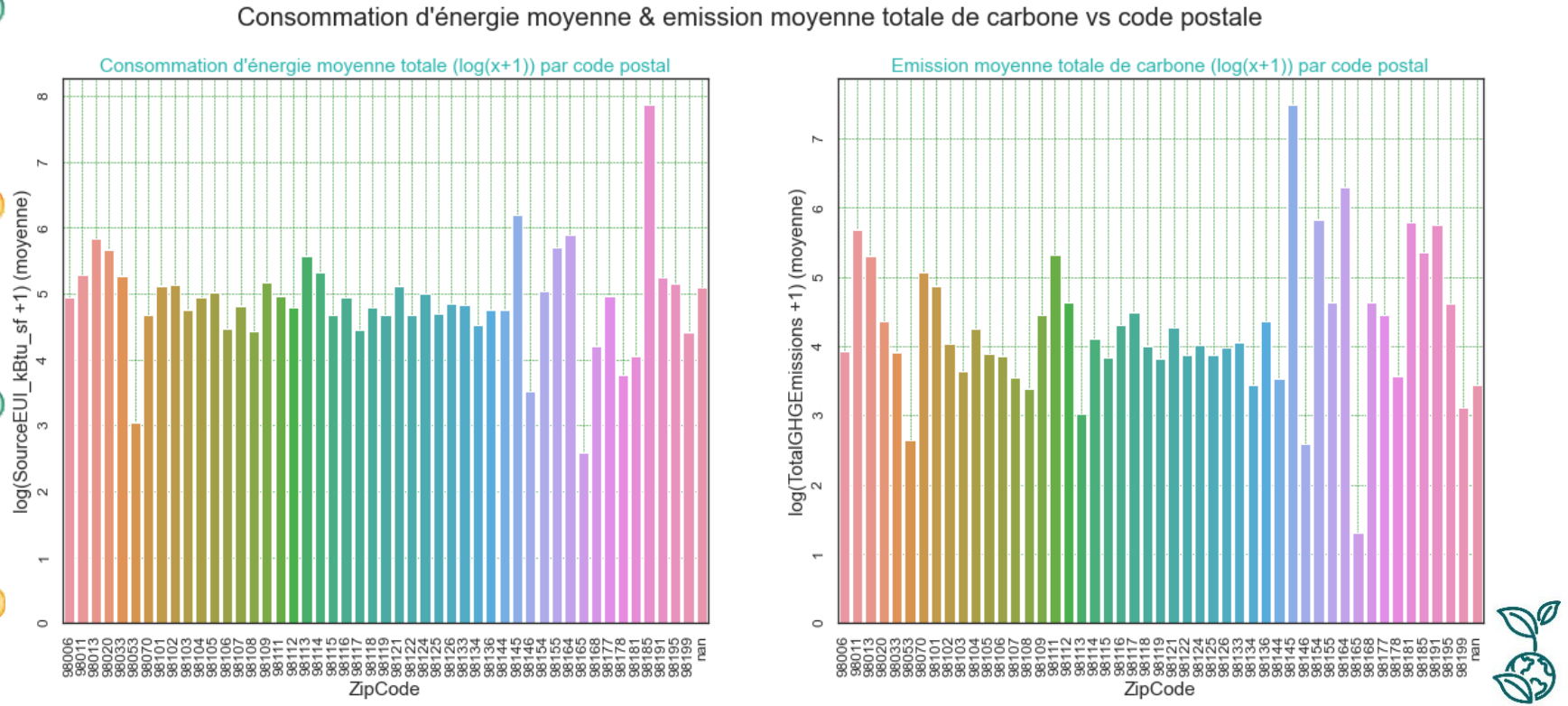
D. Distributions des variables explicatives (features)

Neighborhood: 'Downtow', 'East' et le 'Lake Union' consomment le plus d'énergie et émettent le plus de carbone.



D. Distributions des variables explicatives (features)

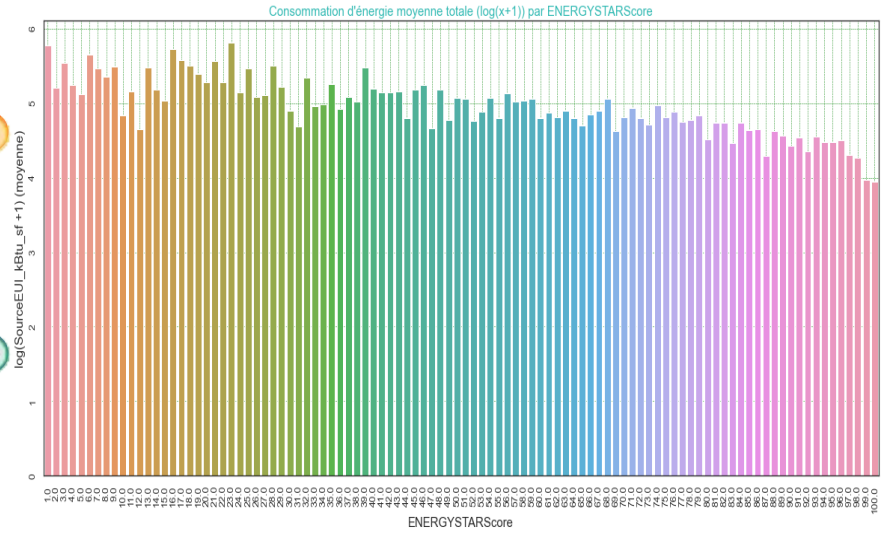
ZipCode: Le code postal 98185 consomme le plus d'énergie.Le code postal 98145 celui émet le plus de carbone.

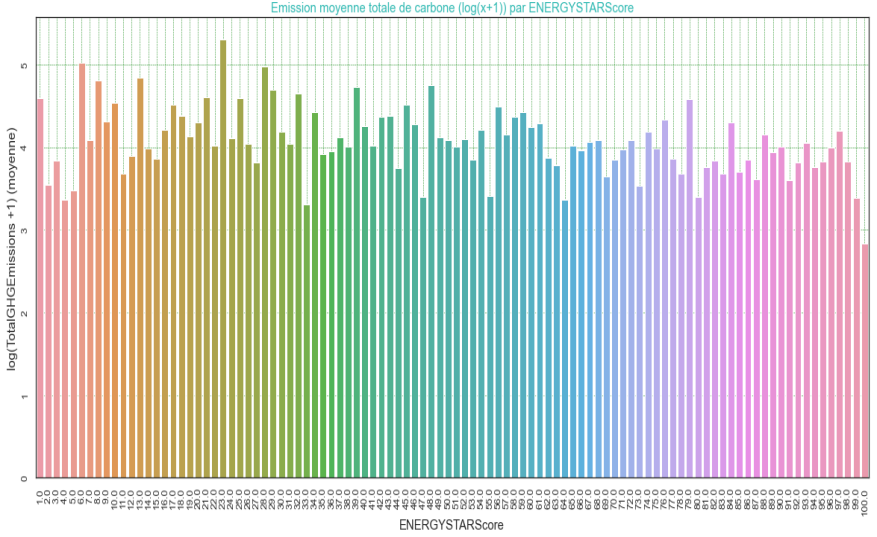


D. Distributions des variables explicatives (features)

ENERGYSTARScore: Pas vraiment de correspondance entre la consommation moyenne d'énergie et le ENERGYSTARScore ; plus de tendance avec l'émission moyenne totale de carbone mais qui ne se confirme pas forcément pour les ENERGYSTARScore bas.

Consommation d'énergie moyenne & emission moyenne totale de carbone vs ENERGYSTARScore





Ceci unis à l'instabilité dans le temps de cette variable, ses difficultés de calcul et le nombre de données manquantes font que ENERGYSTARScore ne soit pas très intéressante pour nous, mais on la testera quand même

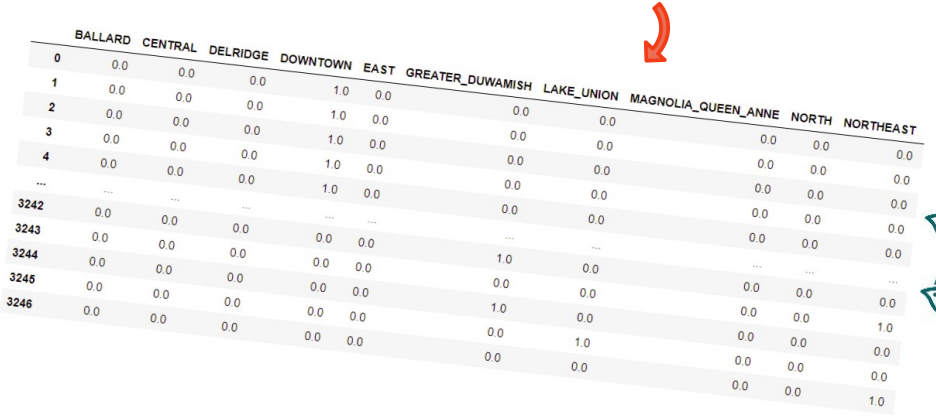
Bases de données FINALES pour nos modèles:

- Deux bases une avec et une sans le **ENERGYSTARScore**
- Features sélectionnés (présents dans les 2 bases):
  - *BuildingType, °YearBuilt, °PropertyGFATotal, *Neighborhood, °ZipCode
- Variables cibles (présents dans les 2 bases):

  °SourceEUI(kBtu/sf) (dépense totale d'énergie), °TotalGHGEmissions (CO2)

'°' → variables numériques
'*' → variables **catégorielles** → on les rend continues → **OneHotEncoder**

	Campus	Multifamily_HR	Multifamily_LR	Multifamily_MR	NonResidential
0	0.0	1.0	0.0	0.0	0.0
1	0.0	1.0	0.0	0.0	0.0
2	0.0	1.0	0.0	0.0	0.0
3	0.0	0.0	1.0	...	...
4	0.0	...	...	0.0	0.0
...	...	1.0	0.0	0.0	0.0
3242	0.0	1.0	0.0	0.0	0.0
3243	0.0	1.0	0.0	0.0	0.0
3244	0.0	1.0	0.0	1.0	0.0
3245	0.0	0.0	0.0		
3246	0.0				



	BALLARD	CENTRAL	DELRIDGE	DOWNTOWN	EAST	GREATER_DUWAMISH	LAKE_UNION	MAGNOLIA_QUEEN_ANNE	NORTH	NORTHEAST
0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...
3242	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
3243	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3244	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0
3245	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
3246	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Procédure à suivre avant de lancer les modèles

- **Training et Test sets.**
  On divise le dataset en 2 sets: un "training set" (80 % des données) et un "test set" (20 % des données).

- **Normalisation des données.**
  Recommandé de normaliser les variables ayant différentes échelles → le modèle peut converger sans normalisation mais l'entraînement est plus difficile et le résultat dépend des unités utilisées pour les variables.

	count	mean	std	min	25%	50%	75%	max
SourceEUI_kBtu_sf_log	2598.0	4.815094	0.894457	0.0	4.360548	4.891476	5.31959	7.828874e+00
YearBuilt	2598.0	1961.527329	32.799983	1900.0	1929.250000	1965.000000	1989.00000	2.015000e+03
PropertyGFATotal	2598.0	115634.953811	262326.888549	16300.0	30152.000000	50017.000000	104313.25000	9.320156e+06
ZipCode_New	2598.0	98116.581601	17.223689	98006.0	98104.000000	98109.000000	98122.00000	9.819900e+04
Campus	2598.0	0.013472	0.115306	0.0	0.000000	0.000000	0.00000	1.000000e+00
Multifamily_HR	2598.0	0.886451	0.317324	0.0	1.000000	1.000000	1.00000	1.000000e+00
Multifamily_LR	2598.0	0.041186	0.198757	0.0	0.000000	0.000000	0.00000	1.000000e+00
Multifamily_MR	2598.0	0.000385	0.019619	0.0	0.000000	0.000000	0.00000	1.000000e+00
NonResidential	2598.0	0.058507	0.234744	0.0	0.000000	0.000000	0.00000	1.000000e+00
BALLARD	2598.0	0.041570	0.199644	0.0	0.000000	0.000000	0.00000	1.000000e+00
CENTRAL	2598.0	0.033872	0.180935	0.0	0.000000	0.000000	0.00000	1.000000e+00
DELRIDGE	2598.0	0.027329	0.163071	0.0	0.000000	0.000000	0.00000	1.000000e+00
DOWNTOWN	2598.0	0.225943	0.418282	0.0	0.000000	0.000000	0.00000	1.000000e+00
EAST	2598.0	0.073133	0.260405	0.0	0.000000	0.000000	0.00000	1.000000e+00
GREATER_DUWAMISH	2598.0	0.200539	0.400481	0.0	0.000000	0.000000	0.00000	1.000000e+00
LAKE_UNION	2598.0	0.090069	0.286336	0.0	0.000000	0.000000	0.00000	1.000000e+00
MAGNOLIA_QUEEN_ANNE	2598.0	0.090839	0.287435	0.0	0.000000	0.000000	0.00000	1.000000e+00
NORTH	2598.0	0.036567	0.187731	0.0	0.000000	0.000000	0.00000	1.000000e+00
NORTHEAST	2598.0	0.071209	0.257223	0.0	0.000000	0.000000	0.00000	1.000000e+00
NORTHWEST	2598.0	0.053888	0.225839	0.0	0.000000	0.000000	0.00000	1.000000e+00
SOUTHEAST	2598.0	0.029638	0.169620	0.0	0.000000	0.000000	0.00000	1.000000e+00
SOUTHWEST	2598.0	0.025404	0.157380	0.0	0.000000	0.000000	0.00000	1.000000e+00

⇦ On normalise les 2 set Training et Test en utilisant uniquement les statistiques sur le "set training"→ les données test et training doivent être projetées dans la même distribution.

On normalise aussi la variable cible → pas vraiment nécessaire, ça évite de possibles bias.

Construction, entraînement et prédiction des modèles.

- **Packaging sklearn:**

1) « .fit », entraîne notre modèle à partir du training set
2) « .predict », prédit la variable cible à partir du test set.
3) « GridSearchCV », lance apprentissage sur des rangs de valeur des hyperparamètres et trouve ceux qui optimisent le modèle.

- **Tout Modèle implique:**

1) Un choix de **métrique** → pour comparer les modèles entre eux.
   Plus courante: R^2, mais un R^2 mauvais (ie. R^2 <<0.6) n'implique pas un modèle inadapté.

   Il faut plusieurs métriques pour évaluer un modèle. Le RMSE (bon si <=0.5), et la MAE (fixée par modèle naïf et à diminuer le plus possible) sont de bonnes métriques de comparaison. On utilisera les 3 pour tous nos modèles.

2) L'optimisation des **hyperparamètres** du modèles (ie. paramètre dont la valeur contrôle le processus d'apprentissage) → en les modulant on contrôle « l'apprentissage» ou adaptation au training set → s'adapter trop au training set rend le test set trop différent et difficile à prédire.

3) Le **temps** d'apprentissage → le moins de temps le mieux.
   Compteur à déclencher juste avant l'optimisation des hyperparamètres, et à stopper juste après l'apprentissage (ie. juste après les fonction 'fit' ou 'GridSearchCV').

R^2 → Coefficient de détermination
RMSE → Erreur quadratique moyenne
MAE → Erreur absolue moyenne

Construction, entraînement et prédiction des modèles.

	Modèle Naïf	Modèles Linéaires		Modèle Non Linéaires	
				Méthode ensembliste	Support Vector Regression
	Dummy Regression (Médiane*)	Linear Regression	ElasticNet	Random Forest Regressor	SVR
Définition	Prédiction basée sur la médiane du training set	Cherche à établir une relation linéaire entre la variable cible et les variables explicatives	Extension de la régression linéaire, avec un paramètre de pénalité supplémentaire qui vise à minimiser la complexité et/ou à réduire le nombre de variables (features) utilisées dans le modèle. Mélange de Lasso (pénalité L1) et Ridge Regression (pénalité L2)	Adapte un certain nombre d'arbre de décision sur divers sous-échantillons de l'ensemble de données et utilise la moyenne pour améliorer la précision prédictive et contrôler le sur ajustement.	Cherche à minimiser les "pénalités" (L2 en particulier), donnant la flexibilité de définir la quantité d'erreur acceptable dans notre modèle et trouvant une ligne appropriée (ou un hyperplan) pour s'adapter aux données.
Hyperparamètres	Pas d'hyperparamètres	Pas d'hyperparamètres	alpha L1_ratio Tol	n_estimators min_samples_leaf max_features Max_depth	gamma Epsilon C
Métriques	R^2 / RMSE / MAE				

L'**optimisation des hyperparamètres** choisis se fait en donnant des **rangs de valeurs à chaque hyperparamètre** et en utilisant ensuite la fonction **« GridSearchCV »** au lieu de « fit » pour lancer l'apprentissage sur tous les rang et ainsi trouver ceux qui optimisent le modèle.

Construction, entraînement et prédiction des modèles.

Hyperparamètres du ElasticNet:

(1 / (2 * n_samples))* ||y - Xw||^2_2 + alpha * l1_ratio * ||w||_1 + 0.5 * alpha * (1 - l1_ratio) * ||w||^2₂

- **alpha:**

  Attribue le poids accordé à chacune des pénalités L1 et L2.

- **L1_ratio:**

  Paramètre de pénalité de mélange.

  L1_ratio = 0 → pénalité L2 (Ridge).

  L1_ratio = 1→ pénalité L1 (Lasso).

  0 < L1_ratio < 1 → pénalité combinaison de L1 et L2.

- **tol:**

  Tolérance d'optimisation → quand le modèle arrête-t-il le processus d'optimisation.

Construction, entraînement et prédiction des modèles.

Hyperparamètres du Random Forest:

- **n_estimators:**
  Nombre d'arbres.

- **min_samples_leaf:**
  Nombre minimum d'échantillons aux feuille de l'arbre.

- **max_features:**
  Nombre de variables (ie. features) à considérer lors de la recherche de la meilleure répartition.

- **max_depth:**
  Module la profondeur pour éviter que le modèle sur apprenne.

Construction, entraînement et prédiction des modèles.

Hyperparamètres du SVR:

- **gamma (Kernel coefficient):**

  Étendue de l'influence d'un seul exemple d'entraînement, valeurs faibles → "loin"; valeurs élevées → "proche".

- **epsilon:**

  Coefficient associé à la pénalité dans la fonction des pertes (loss fonction).

- **C (Coefficient de régularisation):**

  Il calibre le modèle et doit être positif pour éviter des over et under fit.

  Valeurs grandes → marge petite haute précision apprentissage; valeurs petites → grande marge au détriment de l'apprentissage.

Prédiction consommation totale d’énergie

Résultats concernant les dépenses énergétiques

(Métriques pour SourceEUI_kBtu_sf_log)

Sans ENERGYSTARScore

	Modèle	R^2	MSE	RMSE	MAE	Temps de calcul (s)
0	Dummy Regressor	-0.004197	1.035714	1.017700	0.735766	1
1	Linear Regression	0.144916	0.881921	0.939107	0.670013	1
2	ElasticNet	0.143766	0.883107	0.939738	0.668716	15
3	Random Forest	0.528152	0.486657	0.697608	0.400458	316
4	SVR	0.133011	0.894200	0.945621	0.638368	342

Avec ENERGYSTARScore

	Modèle	R^2	MSE	RMSE	MAE	Temps de calcul (s)
0	Dummy Regressor	-0.064185	1.166623	1.080103	0.777485	0
1	Linear Regression	0.329891	0.734613	0.857096	0.559039	0
2	ElasticNet	0.329416	0.735134	0.857399	0.559058	15
3	Random Forest	0.581775	0.458483	0.677114	0.362651	266
4	SVR	0.398244	0.659681	0.812207	0.472230	174

Le meilleur modèle est clairement le Random Forest (considérant ou pas le ENERGYSTARScore).

- Son R^2 atteint presque le 0.6 qui permettrait de valider le modèle.
- Son RMSE reste supérieur à 0.5 mais il a énormément diminué par rapport aux autres modèles.
- Son MAE a lui aussi énormément diminué par rapport au MAE de référence (ie. Celui du modèle naïf Dummy Regressor).

Quant au temps de calcul de l'ordre de 5 min, on peut la considérer bonne compte tenue du nombre de lignes du training set (ie. 2598 lignes sans et 1738 lignes avec le ENERGYSTARScore).

En considérant le **ENERGYSTARScore** on améliore un peu les valeurs des métriques.
Vaut-il le coup de considérer cette variable aux données manquantes et si difficile à calculer? → **NON pas vraiment !**

Importance des variables ("features importances")

(pour le "meilleur" modèle de Random Forest → celui aux "meilleurs" hyperparamètres")

	Coeff
YearBuilt	0.320537
PropertyGFATotal	0.395080
ZipCode_New	0.099425
Campus	0.003465
Multifamily_HR	0.013831
Multifamily_LR	0.007846
Multifamily_MR	0.000040
NonResidential	0.037574
BALLARD	0.009580
CENTRAL	0.009382
DELRIDGE	0.006672
DOWNTOWN	0.011696
EAST	0.006976
GREATER_DUWAMISH	0.023965
LAKE_UNION	0.009186
MAGNOLIA_QUEEN_ANNE	0.010797
NORTH	0.005337
NORTHEAST	0.005324
NORTHWEST	0.007449
SOUTHEAST	0.006749
SOUTHWEST	0.009086

On utilise la fonction « .feature_importances_ » associée au « RandomForestRegressor » de « sklearn.ensemble ».

On ne considère pas la variable ENERGYSTARScore
→ au vu de la performance ça ne vaut pas la peine.

Notez que les variables qui on le plus de poids sont: **YearBuilt** et **PropertyGFATotal**.
Ça ne doit pas nous étonnez compte tenue de nos histogrammes et du poids équitable de chaque classe associée à ces 2 variables.

Prédiction émissions de CO2

Meilleur que pour l'énergie !

Résultats concernant les émissions carbone

(Métriques TotalGHGEmissions_log)

Sans ENERGYSTARScore

	Modèle	R^2	MSE	RMSE	MAE	Temps de calcul (s)
0	Dummy Regressor	-0.004425	1.048740	1.024080	0.792805	1
1	Linear Regression	0.289287	0.742070	0.861435	0.682741	5
2	ElasticNet	0.212837	0.821893	0.906583	0.705609	14
3	Random Forest	0.756289	0.254463	0.504443	0.346645	337
4	SVR	0.396834	0.629777	0.793585	0.618638	457

Avec ENERGYSTARScore

	Modèle	R^2	MSE	RMSE	MAE	Temps de calcul (s)
0	Dummy Regressor	-0.001851	1.018274	1.009096	0.778984	1
1	Linear Regression	0.392084	0.617882	0.786055	0.612551	1
2	ElasticNet	0.390408	0.619585	0.787137	0.613251	25
3	Random Forest	0.752303	0.251757	0.501754	0.358595	411
4	SVR	0.561551	0.445636	0.667560	0.491549	220

Le meilleur modèle est clairement le Random Forest (considérant ou pas le ENERGYSTARScore).

- Son R^2 dépasse le 0.6 qui permettrait de valider le modèle.
- Son RMSE est de l'ordre de 0.5.
- Son MAE a lui aussi énormément diminué par rapport au MAE de référence (ie. Celui du modèle naïf Dummy Regressor).

Quant au temps de calcul de l'ordre de 5 min, on peut la considérer bonne compte tenue du nombre de lignes du training set (ie. 2598 lignes sans et 1738 lignes avec le ENERGYSTARScore).

En considérant le **ENERGYSTARScore** on améliore un peu les valeurs des métriques.
Vaut-il le coup de considérer cette variable aux données manquantes et si difficile à calculer? → **NON pas vraiment !**

Importance des variables ("features importances")

(pour le "meilleur" modèle de Random Forest → celui aux "meilleurs" hyperparamètres")

	Coeff
YearBuilt	0.238204
PropertyGFATotal	0.554891
ZipCode_New	0.084185
Campus	0.012634
Multifamily_HR	0.010432
Multifamily_LR	0.005527
Multifamily_MR	0.000076
NonResidential	0.004293
BALLARD	0.003091
CENTRAL	0.005123
DELRIDGE	0.003354
DOWNTOWN	0.010996
EAST	0.010690
GREATER_DUWAMISH	0.019799
LAKE_UNION	0.005816
MAGNOLIA_QUEEN_ANNE	0.007374
NORTH	0.004313
NORTHEAST	0.005413
NORTHWEST	0.004885
SOUTHEAST	0.003616
SOUTHWEST	0.005288

On utilise la fonction « .feature_importances_ » associée au « RandomForestRegressor » de « sklearn.ensemble ».

On ne considère pas la variable ENERGYSTARScore → au vu de la performance ça ne vaut pas la peine.

Notez que les variables qui on le plus de poids sont: **YearBuilt** et **PropertyGFATotal**.
Ça ne doit pas nous étonnez compte tenue de nos histogrammes et du poids équitable de chaque classe associée à ces 2 variables.

Pour les émissions de CO2 le poids bien supérieur de **PropertyGFATotal** est expliqué au vu de la corrélation de 0.52 existante avec la variable cible **TotalGHGEmissions.**

MERCI