Analyse de Seoul bike dataset

Par Jean-Baptiste ROUSSELLE et Antonin NICOLAS

	Date	Rented Bike Count	Hour	Temperature(°C)	Humidity(%)	Wind speed (m/s)	Visibility (10m)	Dew point temperature(°C)	Solar Radiation (MJ/m2)	Rainfall(mm)	Snowfall (cm)	Seasons	Holiday	Functioning Day
0	01/12/2017	254	0	-5.2	37	2.2	2000	-17.6	0.00	0.0	0.0	Winter	No Holiday	Yes
1	01/12/2017	204	1	-5.5	38	0.8	2000	-17.6	0.00	0.0	0.0	Winter	No Holiday	Yes
2	01/12/2017	173	2	-6.0	39	1.0	2000	-17.7	0.00	0.0	0.0	Winter	No Holiday	Yes
3	01/12/2017	107	3	-6.2	40	0.9	2000	-17.6	0.00	0.0	0.0	Winter	No Holiday	Yes
4	01/12/2017	78	4	-6.0	36	2.3	2000	-18.6	0.00	0.0	0.0	Winter	No Holiday	Yes
5	01/12/2017	100	5	-6.4	37	1.5	2000	-18.7	0.00	0.0	0.0	Winter	No Holiday	Yes
6	01/12/2017	181	6	-6.6	35	1.3	2000	-19.5	0.00	0.0	0.0	Winter	No Holiday	Yes
7	01/12/2017	460	7	-7.4	38	0.9	2000	-19.3	0.00	0.0	0.0	Winter	No Holiday	Yes
8	01/12/2017	930	8	-7.6	37	1.1	2000	-19.8	0.01	0.0	0.0	Winter	No Holiday	Yes
9	01/12/2017	490	9	-6.5	27	0.5	1928	-22.4	0.23	0.0	0.0	Winter	No Holiday	Yes

Explication du problème

- La problématique choisie est : **comment prédire la variable “Rented Bike Count” en fonction des autres variables du dataset ?**

  C’est à dire, prédire le nombre de vélos loués en fonction de l’heure, la température, la vitesse du vent, …

- Cette question est intéressante à nos yeux car pour y répondre, nous allons devoir prédire un comportement humain (prendre le vélo) en fonction d’éléments extérieurs (les conditions climatiques). De plus, ces prédictions pourrait par la suite être utilisé par une entreprise de location de vélo à Séoul.

Analyse du problème

- A ce stade de la résolution de la question, nous somme déjà convaincu que les conditions climatiques (informations dans la base) influent sur l'utilisation des vélos. Ainsi nous espérons pouvoir modéliser cette corrélation avec un modèle d'apprentissage supervisé vu en cours.
- A première vue, la résolution de cette question devrait permettre d'exploiter presque toutes les caractéristiques (colonnes) de la base de données.

Data cleaning : 1ère partie

- Suppression de la colonne “Functionning Day” dont les valeurs sont des booléens, car la variable“Rented Bike Count” vaut toujours 0 quand “Functionning Day” vaut False
- Suppression de la colonne “Date” car l’heure et la saison sont déjà stocké dans d’autre colonne, il y a donc redondance. De plus, ces informations nous sembles plus exploitable que la date.

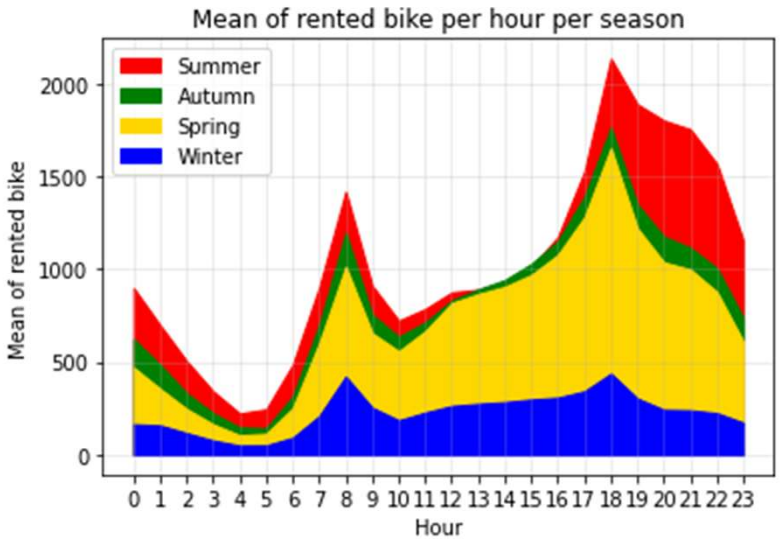
Data cleaning : 2ème partie

Pour faciliter l'exploitation des données, nous réalisons les changements suivants :

- on convertit la colonne "Holiday" en booléen car elle est composé de 2 valeurs
- on remplace les noms de saisons par des nombres
- on s'assure qu'il n'y pas de valeurs NaN ou de string vide

Data Visualisation : 1ère partie

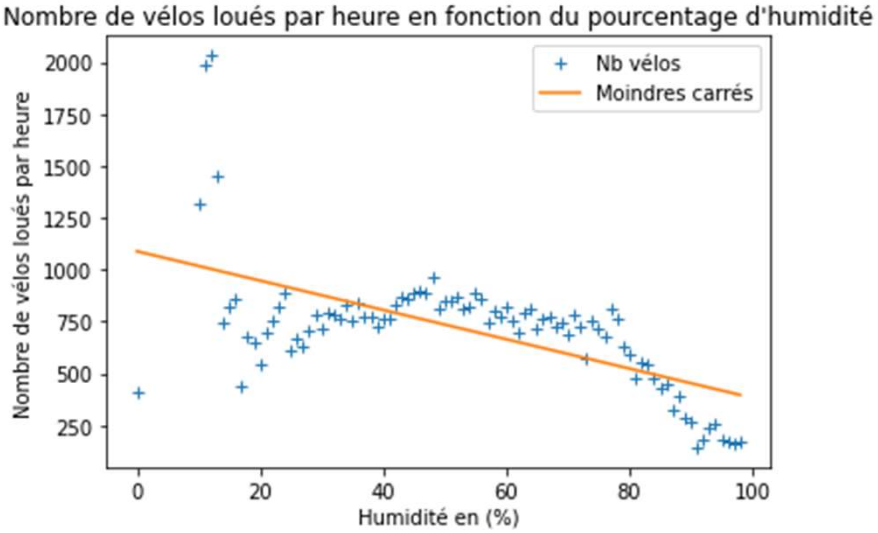
Pour s'assurer de l'impact de chaque variable sur "Rented Bike Count" nous avons réaliser des graphiques



Ce graphique explicite clairement la corrélation entre l'heure et le nombre de vélos loués. (plus de location vers 8h et 18h, les heures de départ et de retour au travaille)

En même temps, on peut en déduire que la saison influe bien sur le nombre de vélos loués. (plus de location en été qu'en hiver)

Data Visualisation : 2ème partie

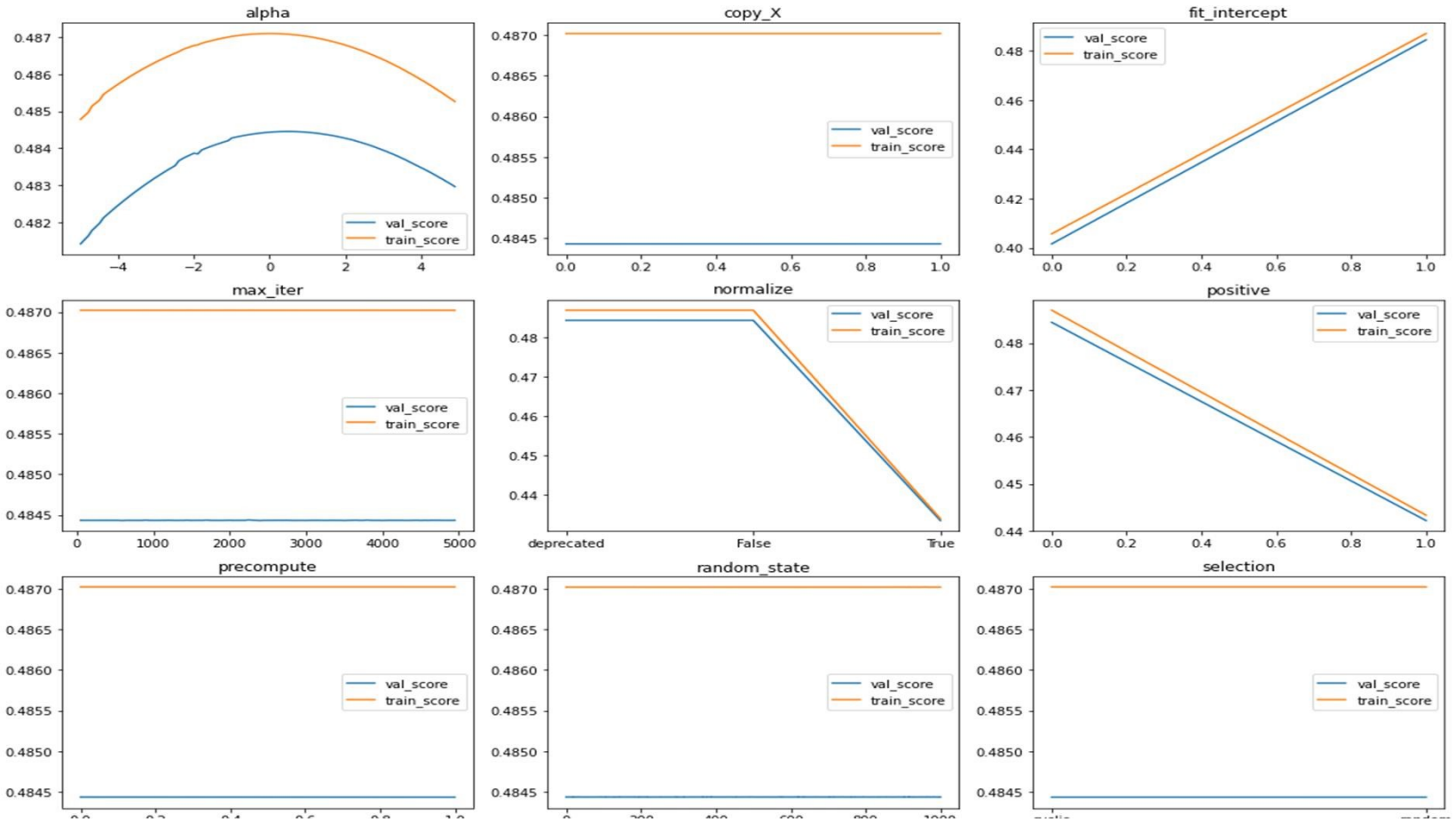


On voit grâce à la droite des moindre carrés que le nombre de vélos diminue avec l'augmentation de l'humidité. Ainsi l'humidité est directement corrélé au nombre de location.

Modélisation : 1ère partie

- Séparation de la base de donnée en deux jeux de données, un pour l'entraînement et un pour le test
- Standardisation des jeux de données
- Création d'une fonction affichant la prédiction d'un modèle en fonction de la variation de chacun de ses paramètres (appelée validation_curves())
- Nous avons à faire à un problème de régression car la prédictions concerne le nombre de vélos loués.

Modèle Lasso : exemple de sortie de validation_curves()



Modélisation : 3ème partie : prédictions obtenues

- Pour le modèle **Lasso**, on obtient **0.466**
- Pour le modèle **Régression Linéaire**, on obtient **0.465**
- Pour le modèle **Arbre de Décision**, on obtient **0.643**
- Pour le modèle **K Plus Proche Voisins**, on obtient **0.69**
- Pour le modèle **Random Forest**, on obtient **0.755**