

Seoul Bike Demand



PROBLÉMATIQUE

Le data set Seoul Bike Sharing Demand est un jeu de données contenant le nombre de vélos publics loués à chaque heure dans le système de vélos en libre-service de Séoul, contenant les données météorologiques et les informations sur les vacances correspondantes.

Notre objectif étant de pouvoir produire un modèle de prédiction du nombre de vélos loués en fonction des conditions météorologiques mesurées.

PROBLÉMATIQUE

Enjeux:

- Santé
- Pollution
- Transport / diminuer le nombre de voiture
- Adaptation à la demande

Pour cela, comment nous est-il possible d'anticiper la demande du nombre de vélos loués en fonction des conditions météorologiques?

Data-Exploring

Le dataset contient des informations météorologiques (température, humidité, vitesse du vent, visibilité, point de rosée, rayonnement solaire, chutes de neige, précipitations), le nombre de vélos loués par heure et des informations de date.

source :<https://archive.ics.uci.edu/ml/datasets/Seoul+Bike+Sharing+Demand#>

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
01/12/2017	254	0	-5.2	37	2.2	2000	-17.6	0.0	0.0	0.0	Winter	No Holiday	Yes
01/12/2017	204	1	-5.5	38	0.8	2000	-17.6	0.0	0.0	0.0	Winter	No Holiday	Yes
01/12/2017	173	2	-6.0	39	1.0	2000	-17.7	0.0	0.0	0.0	Winter	No Holiday	Yes
01/12/2017	107	3	-6.2	40	0.9	2000	-17.6	0.0	0.0	0.0	Winter	No Holiday	Yes
01/12/2017	78	4	-6.0	36	2.3	2000	-18.6	0.0	0.0	0.0	Winter	No Holiday	4 Yes

Data-Exploring

Explorons le dataset :

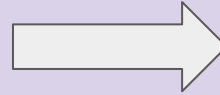
- les colonnes
 - la taille du dataset
 - les types
- => majorité du dataset en float et int
- + vérification du dataset propre

```
>>> <class 'pandas.core.frame.DataFrame'>
RangeIndex: 8760 entries, 0 to 8759
Data columns (total 14 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Date                                  8760 non-null   object
1   Rented Bike Count                    8760 non-null   int64
2   Hour                                 8760 non-null   int64
3   Temperature(°C)                     8760 non-null   float64
4   Humidity(%)                         8760 non-null   int64
5   Wind speed (m/s)                    8760 non-null   float64
6   Visibility (10m)                    8760 non-null   int64
7   Dew point temperature(°C)           8760 non-null   float64
8   Solar Radiation (MJ/m2)             8760 non-null   float64
9   Rainfall(mm)                       8760 non-null   float64
10  Snowfall (cm)                      8760 non-null   float64
11  Seasons                             8760 non-null   object
12  Holiday                             8760 non-null   object
13  Functioning Day                     8760 non-null   object
dtypes: float64(6), int64(4), object(4)
memory usage: 958.2+ KB
```

Data-PreProcessing

- On a séparé la date en année, mois et jour afin que nous puissions comparer plus facilement par mois et jour de la semaine.

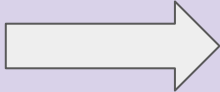
Date
01/12/2017
01/12/2017
01/12/2017
01/12/2017
01/12/2017



Year	Month	Weekday
2017	December	Friday
2017	December	Friday
2017	December	Friday
2017	December	Friday
2017	December	Friday

Data-PreProcessing

- On a défini une colonne **jour et nuit** en encodant la colonne heure
- Puis on va séparer les colonnes en **deux types**: les numériques et les catégoriques pour procéder plus facilement à la visualisation du dataset

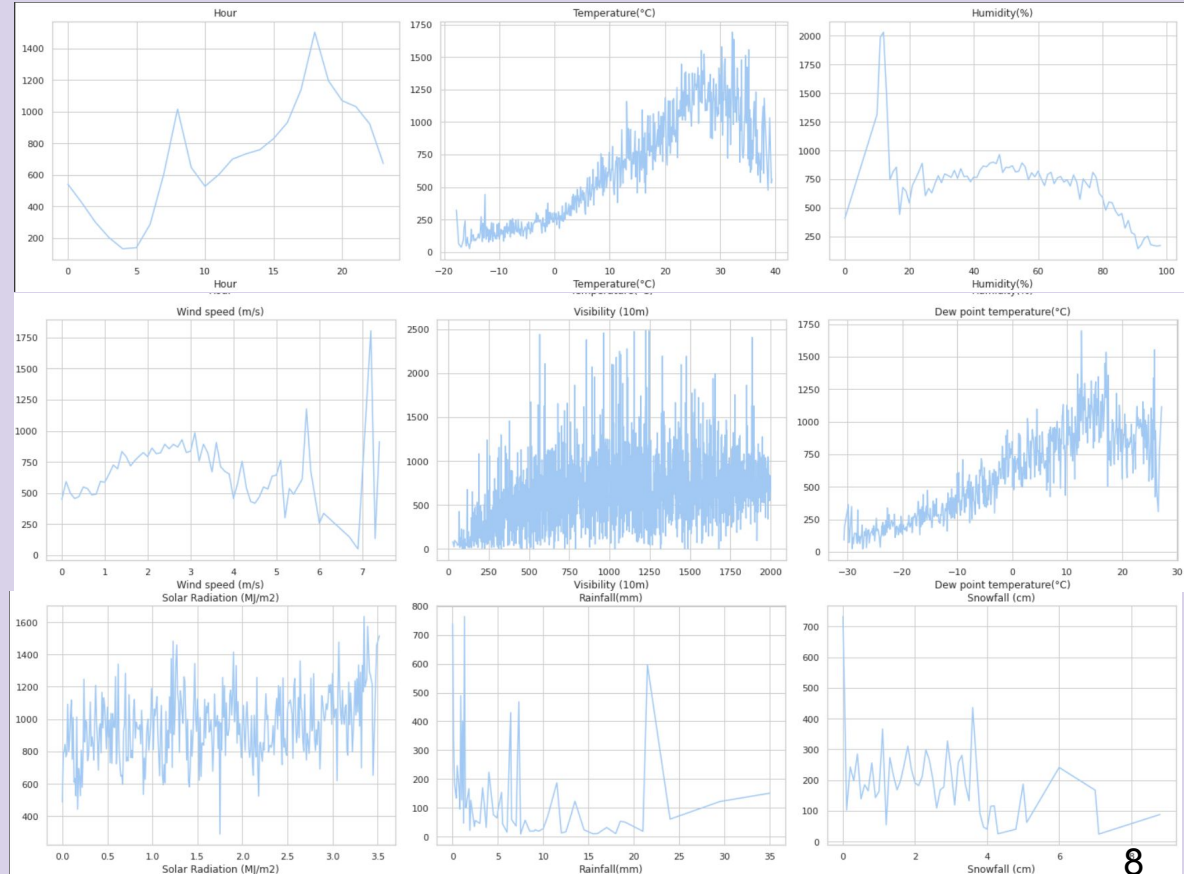


Hour	Daytime
0	Night
1	Night
2	Night
3	Night
4	Night

Data-Visualisation

- Visualisation globale du dataset numérique

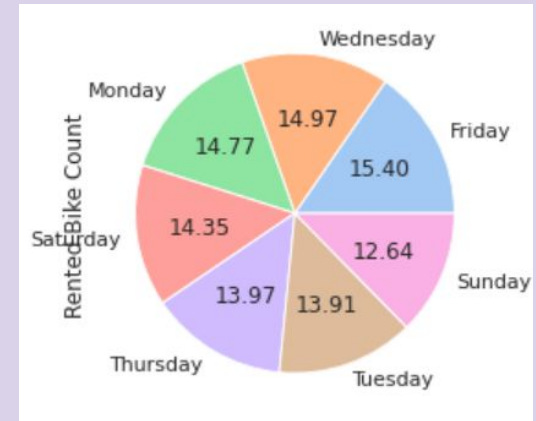
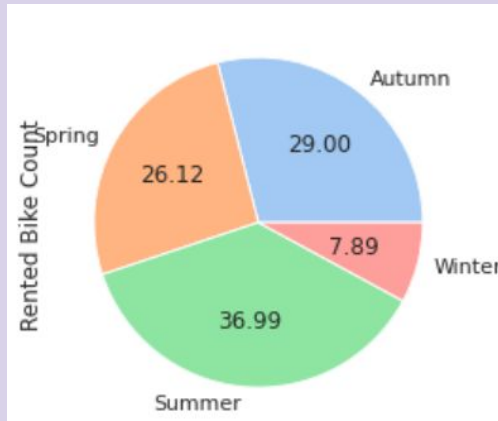
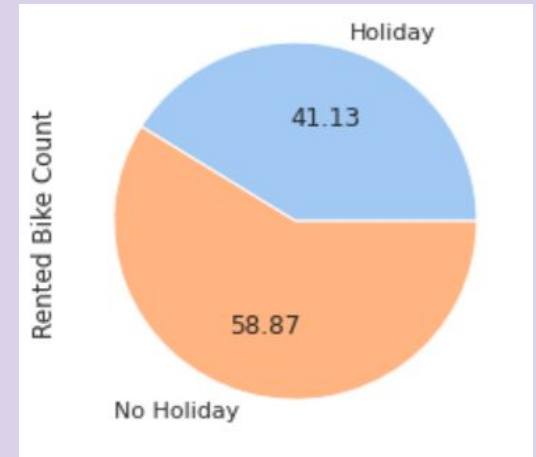
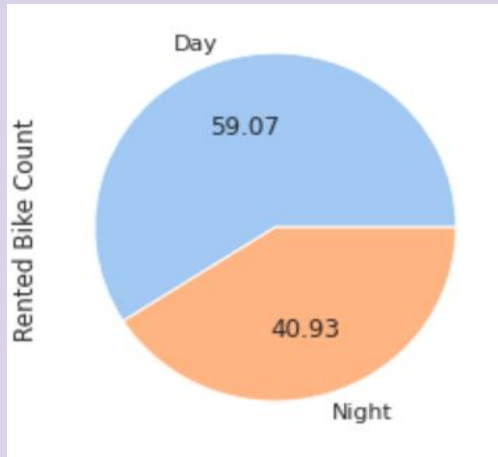
Pour chaque valeur de chaque variable, on étudie l'évolution moyenne du nombre de vélos loués.



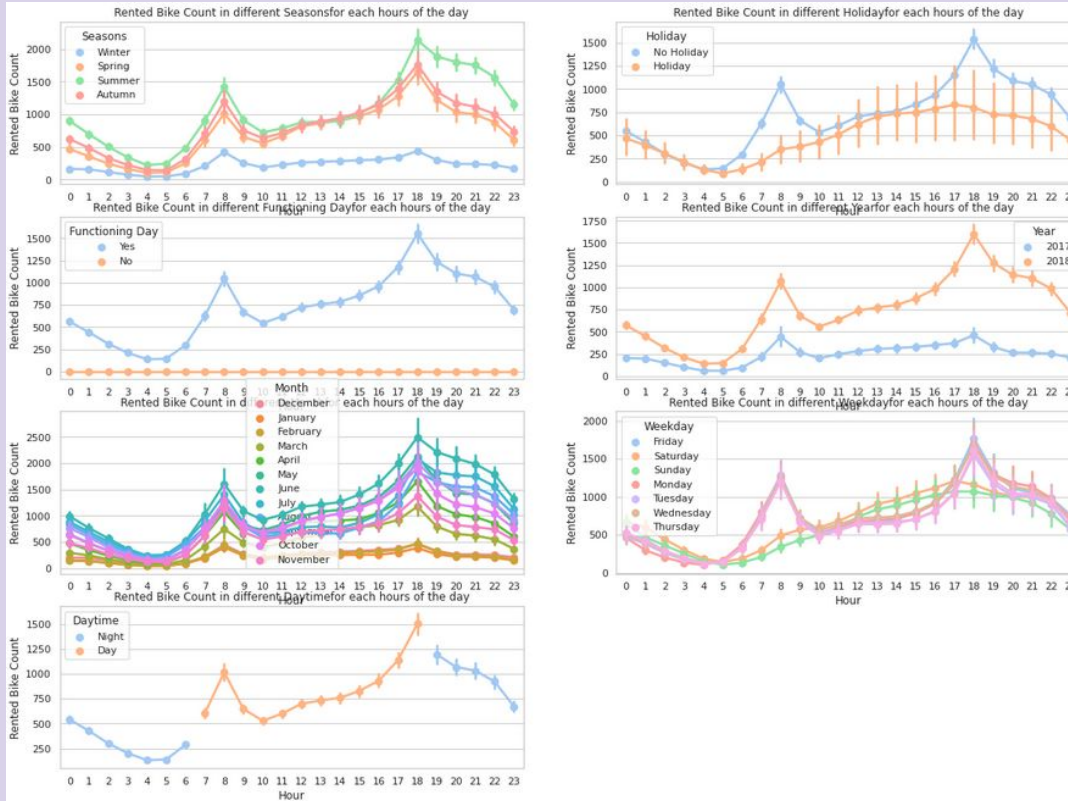
Data-Visualisation

- Visualisation globale du dataset par catégorie

Pour chaque valeur de chaque variable, on étudie la proportion du nombre de vélos loués.

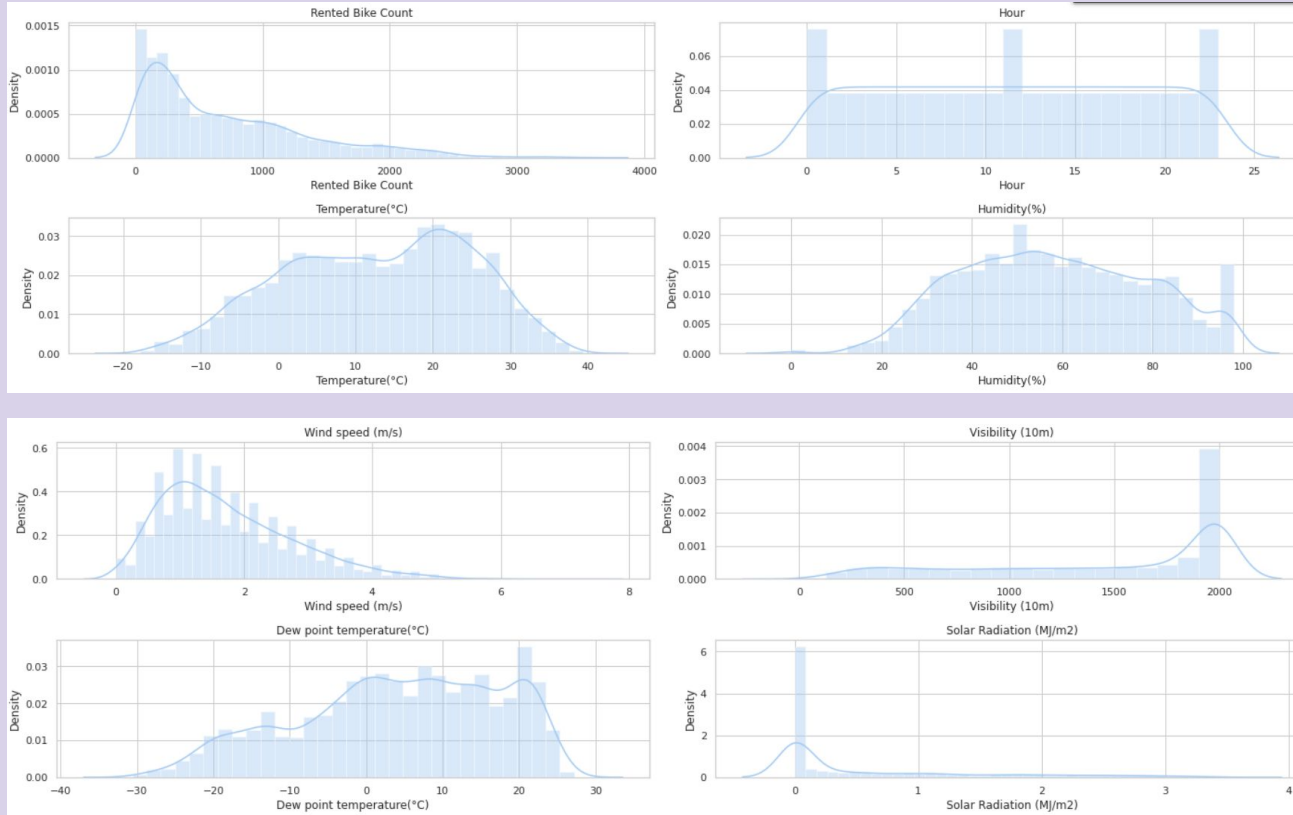


Data-Visualisation



On a affiché pour chaque heure de la journée et pour chaque catégorie, le nombre moyen de vélos loués.

Data-Visualisation

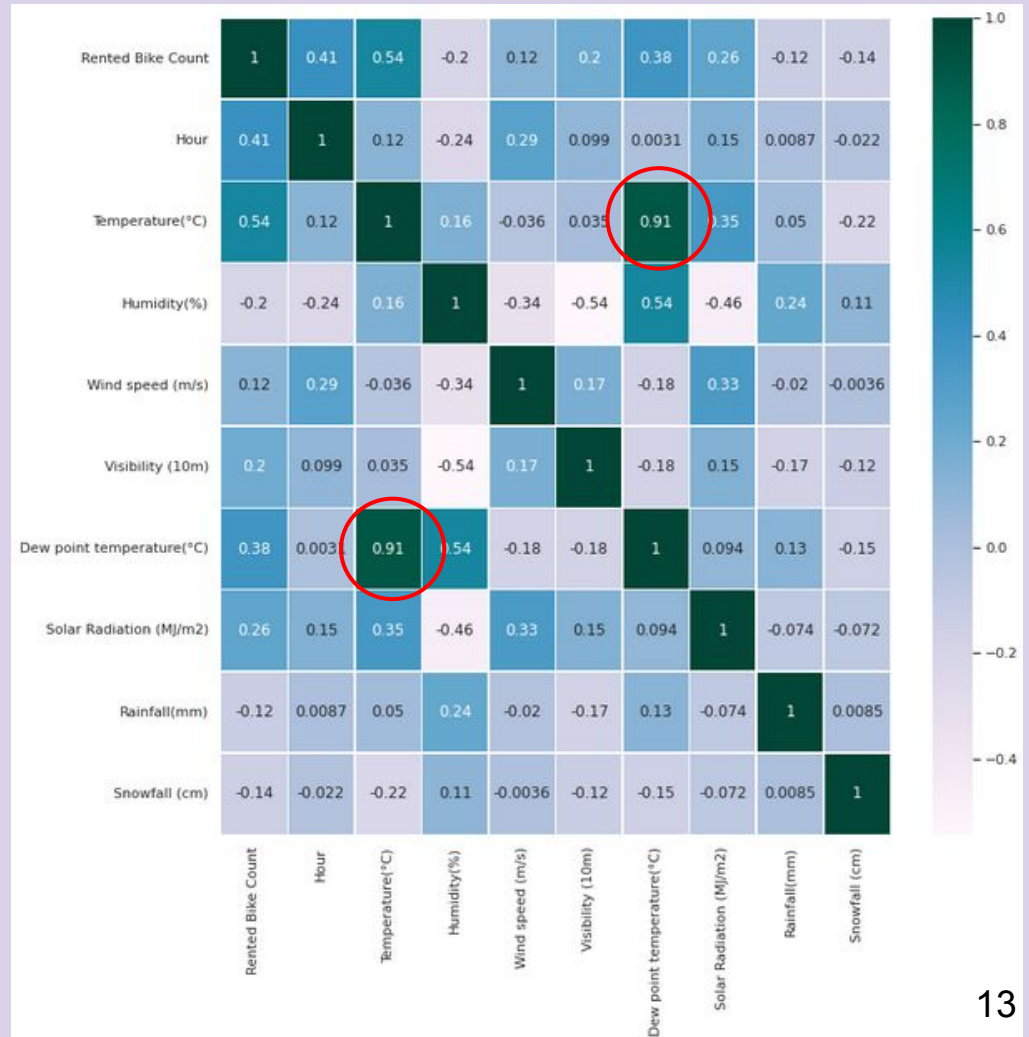


Pour chaque valeur de chaque variable, on vérifie la répartition des données numériques afin qu'on puisse le standardiser plus tard.

Data-Visualisation

- Corrélation des données

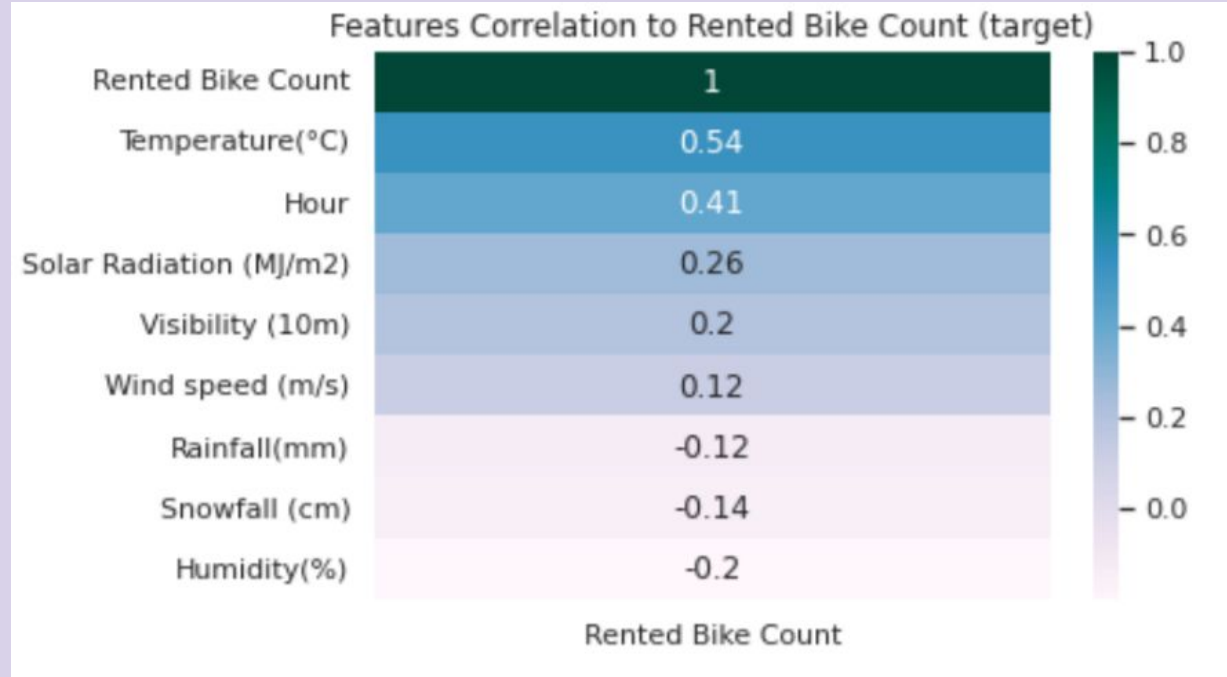
Déduction: grosse corrélation entre température et dew point



Data-Visualisation

- Corrélation des données avec la cible.

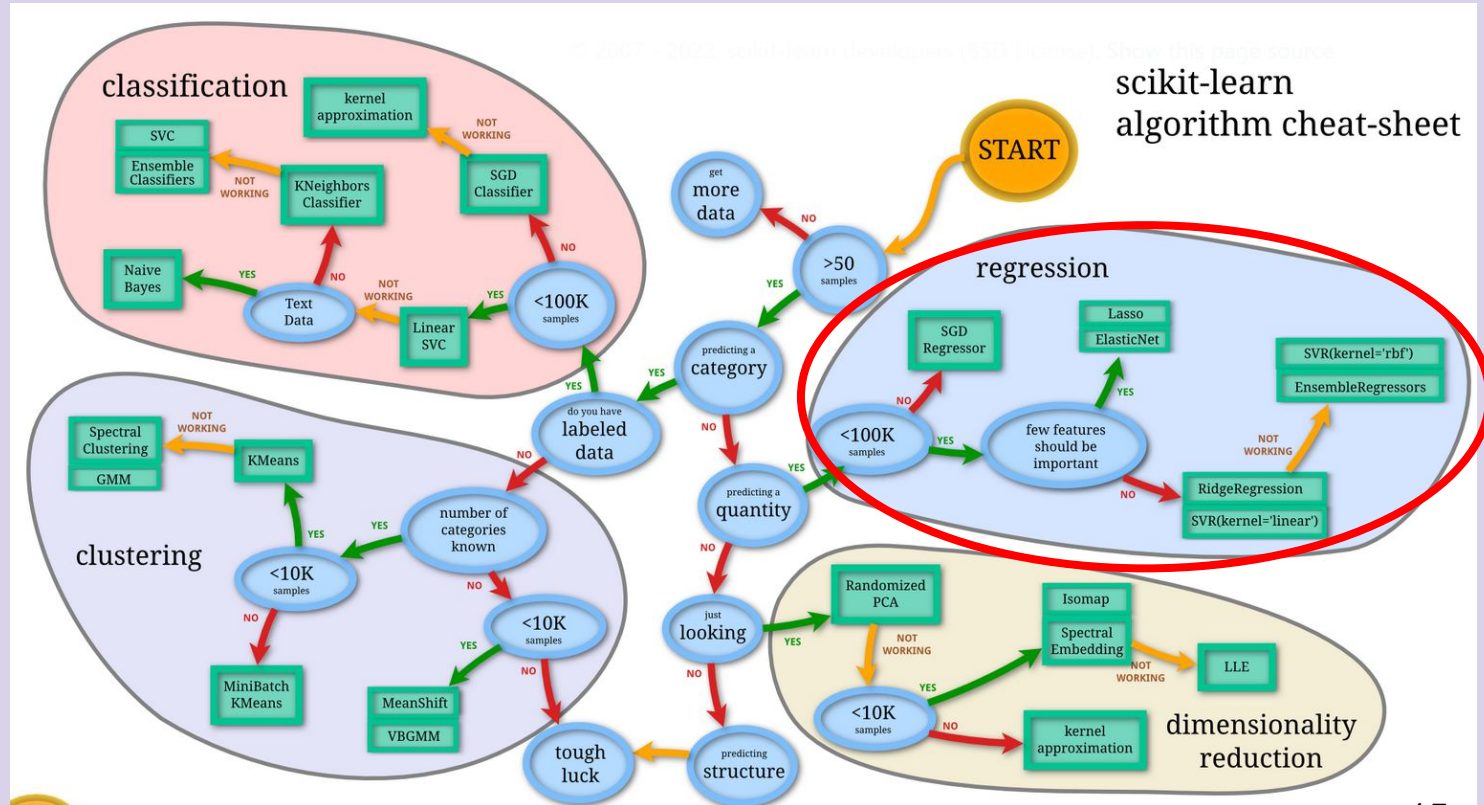
Déduction: La température, l'heure et l'ensoleillement



Data-Modeling

Choix de model :

- modéliser le nombre de vélos loués par heure
- un cas où nous devons utiliser la régression



Data-Modeling

Pour modéliser notre cible, nous utiliserons plusieurs types de régressions puisque nous souhaitons modéliser le nombre de vélos loués par heure.

On utilisera alors l'indicateur R^2 pour comparer nos modèles.

Tout d'abord, nous allons séparer le dataset en : train et test .

Puis, nous normaliserons les données.

- 1) *Regression linéaire*
- 2) *SVR Regression*
- 3) *Decision Tree (Regression)*
- 4) *Random Forest (Regression)*
- 5) *RIDGE*
- 6) *Lasso*
- 7) *KNN*

Data-Modeling

1) Régression Linéaire Multiple

The model performance for testing set

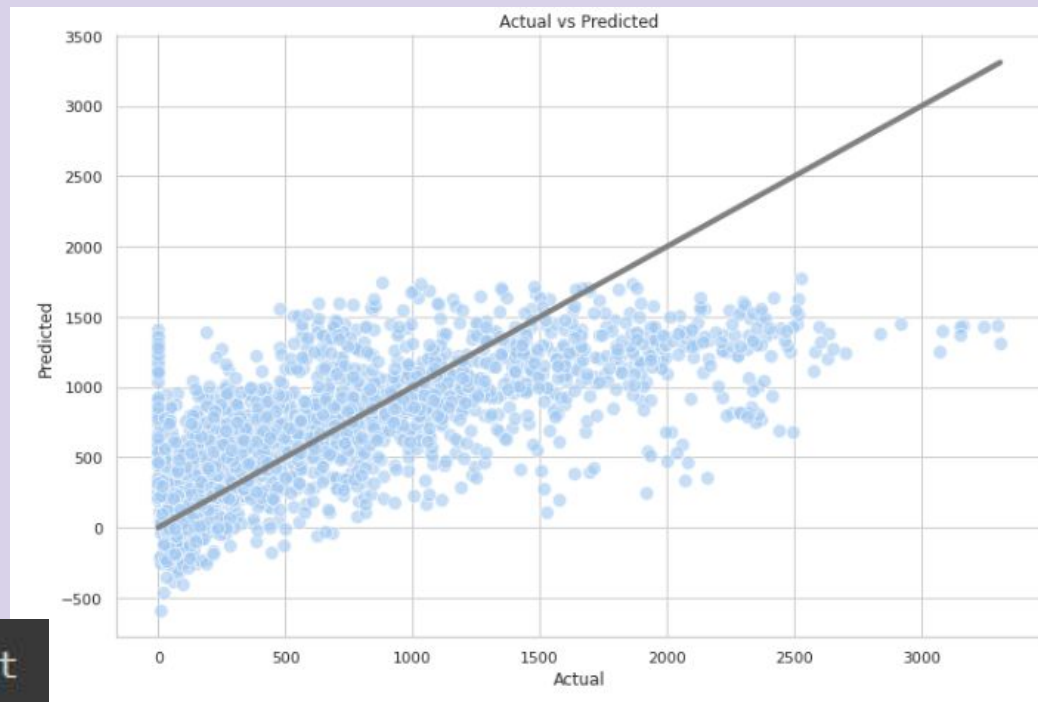
 R^2 : 0.47831423177330046

Adjusted R^2 : 0.4764006663694428

MAE: 351.99180856038186

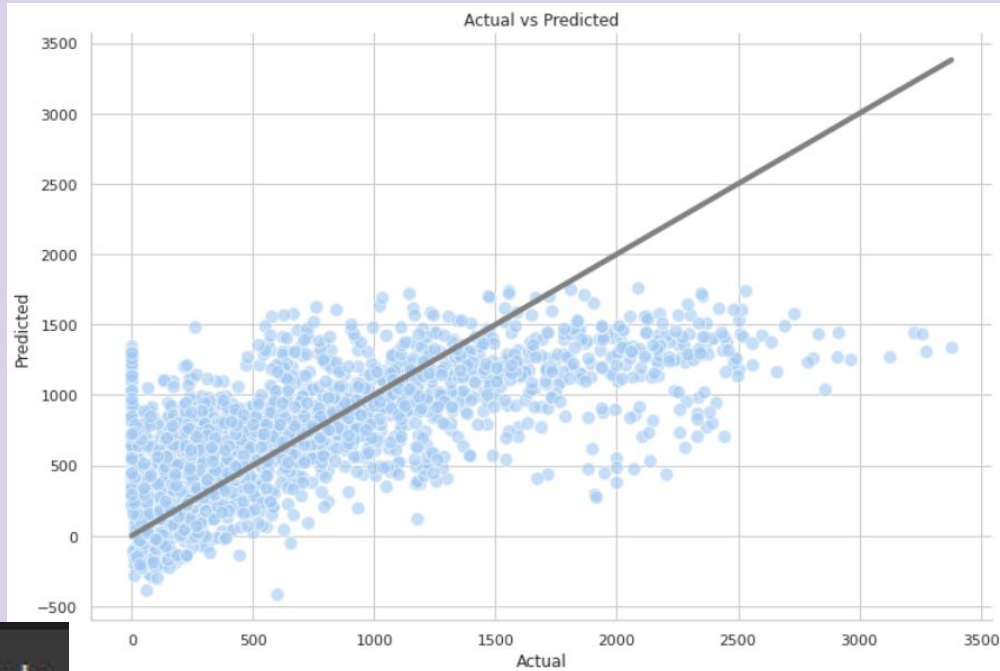
MSE: 225035.74153749936

RMSE: 474.37932241772444



Data-Modeling

2) Ridge Regression



The model performance for testing set

 R^2 : 0.47824956295900556

Adjusted R^2 : 0.47633576034720915

MAE: 351.96834910231075

MSE: 225063.63724688138

RMSE: 474.4087238309192

Data-Modeling

3) Lasso Regression

The model performance for testing set

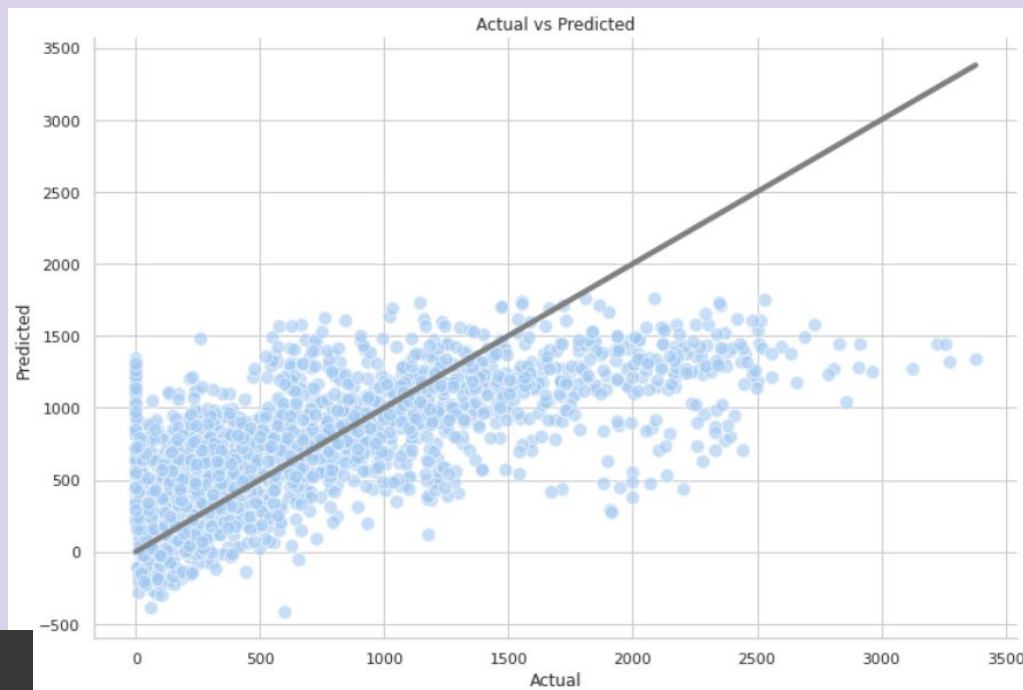
 R^2 : 0.47831423177330046

Adjusted R^2 : 0.4764006663694428

MAE: 351.99180856038186

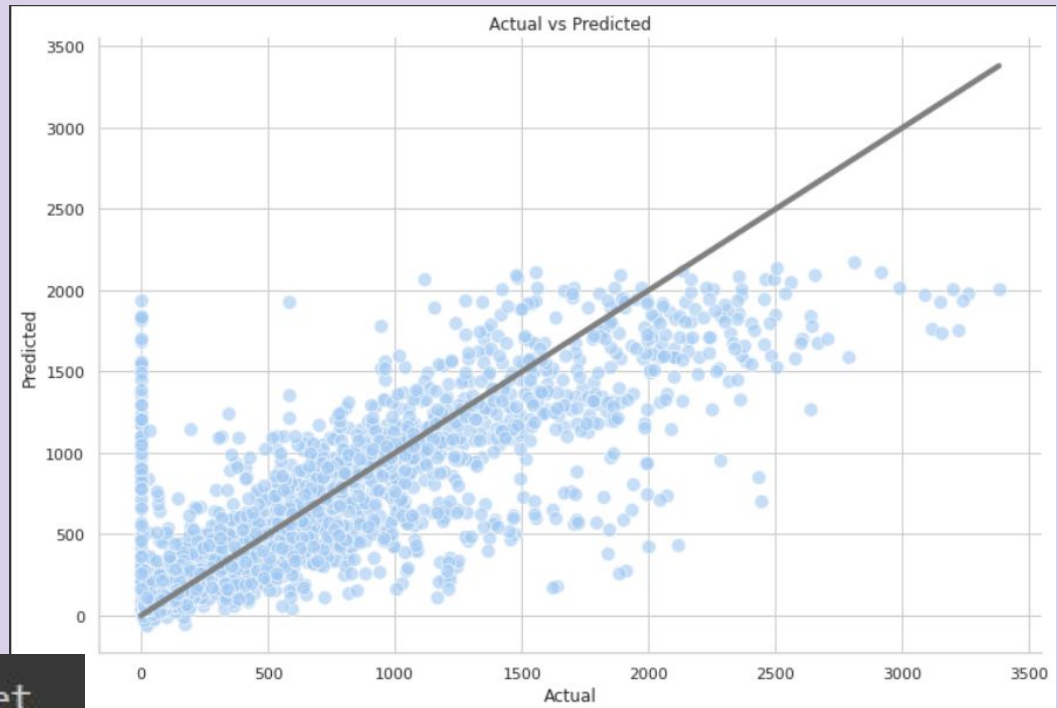
MSE: 225035.74153749933

RMSE: 474.3793224177244



Data-Modeling

4) Régression SVR (rbf)



The model performance for testing set

 R^2 : 0.6447455327690399

Adjusted R^2 : 0.6434424443977205

MAE: 242.007763437621

MSE: 153243.4989353364

RMSE: 391.46327916592173

Data-Modeling

5) KNN Regressor

The model performance for testing set

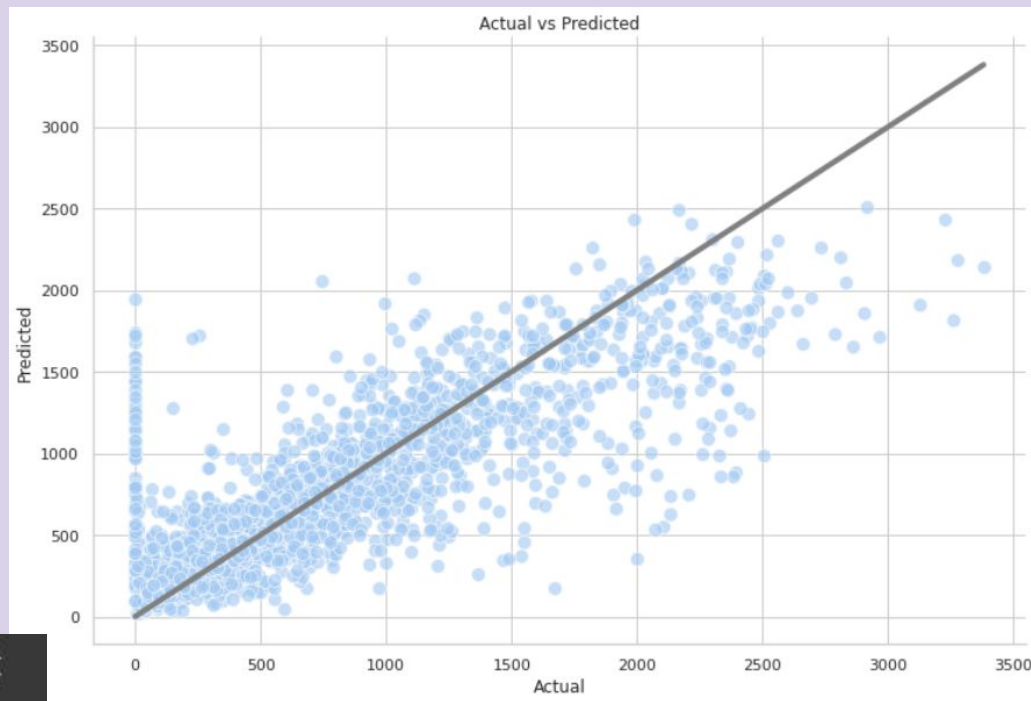
 R^2 : 0.7065529612968362

Adjusted R^2 : 0.7054765851805477

MAE: 227.84893455098936

MSE: 126582.08442415018

RMSE: 355.78376076508914



Data-Modeling

6) Decision Tree Regression

The model performance for testing set

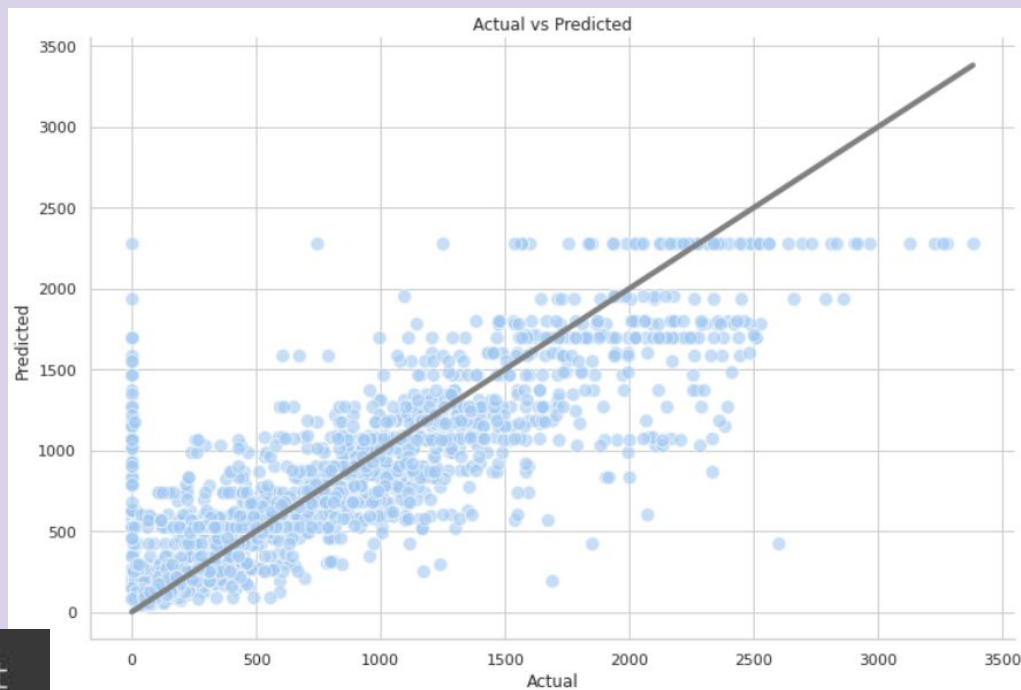
 R^2 : 0.7366093758456665

Adjusted R^2 : 0.7356432479257974

MAE: 218.01616253932642

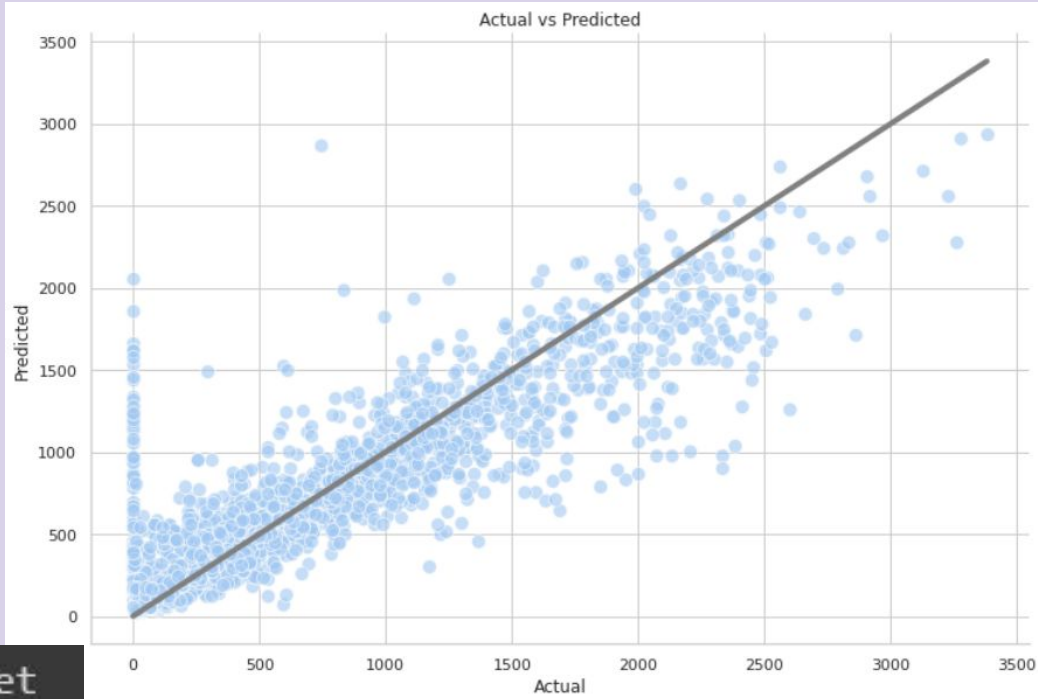
MSE: 113616.87059639762

RMSE: 337.07101714089515



Data-Modeling

7) Random Forest Regression



The model performance for testing set

 R^2 : 0.7815795508371141

Adjusted R^2 : 0.7807783754160672

MAE: 189.23468493150682

MSE: 94218.41794036528

RMSE: 306.9501880441927

Data-Modeling

Résultat

	Model	R_2	Adjusted_R_2
3	Regression Random Forest	0.761542	0.760667
2	Regression Decision Tree	0.700026	0.698926
6	KNN	0.676490	0.675303
1	SVR	0.624268	0.622890
5	Lasso	0.452149	0.450140
4	Ridge	0.452075	0.450065
0	Linear Model	0.452001	0.449991

	Model	R_2	Adjusted_R_2	MAE	MSE	RMSE
3	Regression Random Forest	0.761542	0.760667	184.077164	97409.753241	312.105356
2	Regression Decision Tree	0.700026	0.698926	215.275800	122538.525997	350.055033
6	KNN	0.676490	0.675303	232.722933	132153.182688	363.528792
1	SVR	0.624268	0.622890	241.806310	153485.500743	391.772256
5	Lasso	0.452149	0.450140	351.195052	223795.778794	473.070585
4	Ridge	0.452075	0.450065	351.491680	223826.026916	473.102554
0	Linear Model	0.452001	0.449991	351.620958	223856.251259	473.134496

API avec Flask

Etapes :

> Importer l'environnement **venv**

> Activer l'environnement **venv**

> Lancer python app.py

> `http://127.0.0.1:5000`

The screenshot shows a web application titled "Prediction du nombre de vélos loués". It features a vertical list of input fields for various weather and time-related factors: "Hour", "Temperature(°C)", "Humidity(%)", "Wind speed (m/s)", "Visibility (10m)", "Solar Radiation (MJ/m2)", "Rainfall(mm)", and "Snowfall (cm)". The "Humidity(%)" field has a "Score" label next to it. At the bottom of the form is a blue "Predict" button. A vertical scrollbar is visible on the right side of the input fields.

Exemple :

Hour	10
Temperature(°C)	18.1
Humidity(%)	46
Wind speed (m/s)	2.9
Visibility (10m)	1755
Solar Radiation (MJ/m2)	2.17
Rainfall(mm)	0.0
Snowfall (cm)	0.0

Rented Bike Count	877
-------------------	-----

La prédiction du nombre de vélos loués
est : 835.0

Conclusion



Avec cette étude sur la location de vélos à Séoul, cela nous a permis de répondre à la problématique en choisissant un modèle approprié pour l'anticipation de la demande du nombre de vélos loués en fonction des conditions météorologiques.

Enfin, une perspective de future est qu'on puisse réaliser le même modèle à l'échelle internationale adaptés aux grandes métropoles comme Paris qui aujourd'hui se transforme de plus en plus en une ville cyclable et de mieux prévoir la demande.