SEOUL BIKE DATA

Students :
Théo PANTECOUTEAU
Hugo SEQUIER

TABLE OF CONTENTS

01

INTRODUCTION

Ins and outs of the subject

A BIT OF CONTEXT

BIKE SHARES IN SEOUL

Our project focus on predicting the number of bike shares in Seoul based on meteorological data.

PRACTICAL APPLICATION

Urban Planning
Traffic management
Supporting the growth of eco-friendly transportation methods

THE PROBLEM IN THE STUDY CONTEXT

DATA SCIENCE — Data cleaning, visualization, predictive model and how to deploy it

REAL-WORLD — Real-world issues, offering insights into sustainable urban transportation

INTERDISCIPLINARY — Urban planning, environmental science, data analytics and how data-driven approaches can inform business decisions

02

PROJECT MANAGEMENT

How did we work?

HOW DID WE WORK?

GITHUB

Facilitates collaboration, allowing to work simultaneously without overriding each other's work.

BRANCH

Allow us to work on different features or fixes in isolation, without affecting the main or 'master' branch.

03

DATA PROCESSING

Data preparation, data visualization and modeling

DATA PRE-PROCESSING / 1

```
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
```

```
Date                       0
Rented Bike Count          0
Hour                       0
Temperature(°C)            0
Humidity(%)                0
Wind speed (m/s)           0
Visibility (10m)           0
Dew point temperature(°C)  0
Solar Radiation (MJ/m2)    0
Rainfall(mm)               0
Snowfall (cm)              0
Seasons                    0
Holiday                    0
Functioning Day            0
```

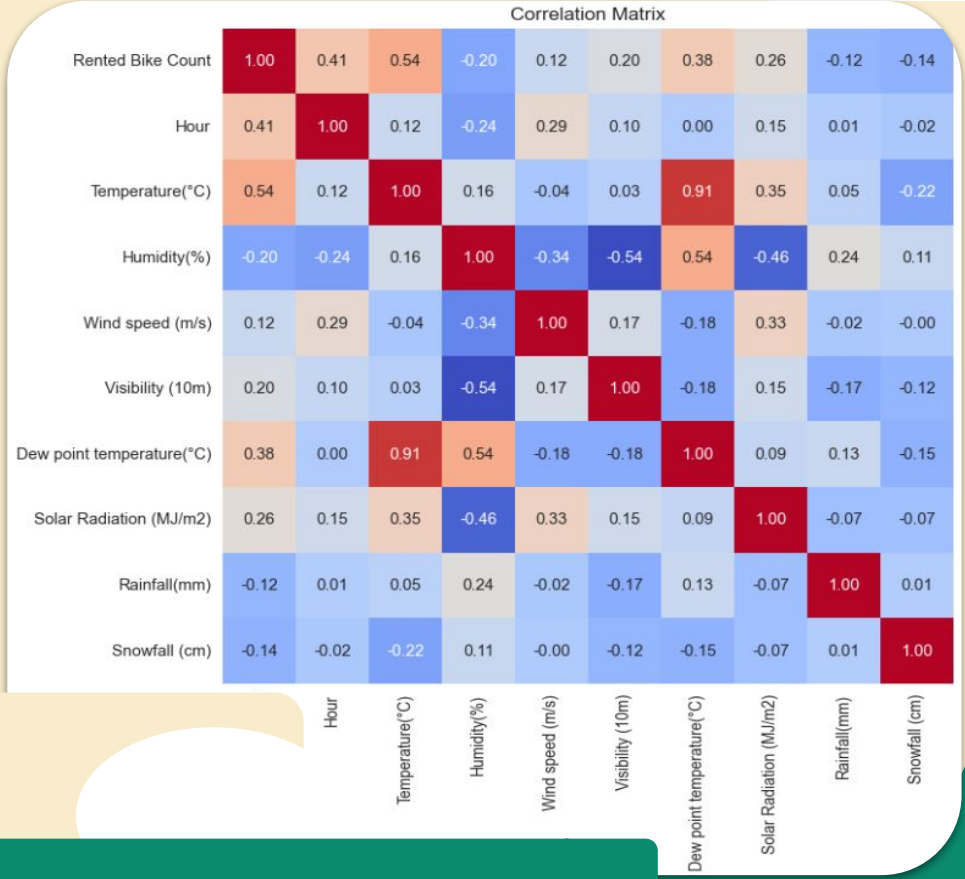
	count	mean	std	min	25%	50%	75%	max
Rented Bike Count	8760.0	704.602055	644.997468	0.0	191.00	504.50	1065.25	3556.00
Hour	8760.0	11.500000	6.922582	0.0	5.75	11.50	17.25	23.00
Temperature(°C)	8760.0	12.882922	11.944825	-17.8	3.50	13.70	22.50	39.40
Humidity(%)	8760.0	58.226256	20.362413	0.0	42.00	57.00	74.00	98.00
Wind speed (m/s)	8760.0	1.724909	1.036300	0.0	0.90	1.50	2.30	7.40
Visibility (10m)	8760.0	1436.825799	608.298712	27.0	940.00	1698.00	2000.00	2000.00
Dew point temperature(°C)	8760.0	4.073813	13.060369	-30.6	-4.70	5.10	14.80	27.20
Solar Radiation (MJ/m2)	8760.0	0.569111	0.868746	0.0	0.00	0.01	0.93	3.52
Rainfall(mm)	8760.0	0.148687	1.128193	0.0	0.00	0.00	0.00	35.00
Snowfall (cm)	8760.0	0.075068	0.436746	0.0	0.00	0.00	0.00	8.80

Object Values which we'll need to encode
Normalize to ensure consistency in interpretation
No missing values
Keeping outliers

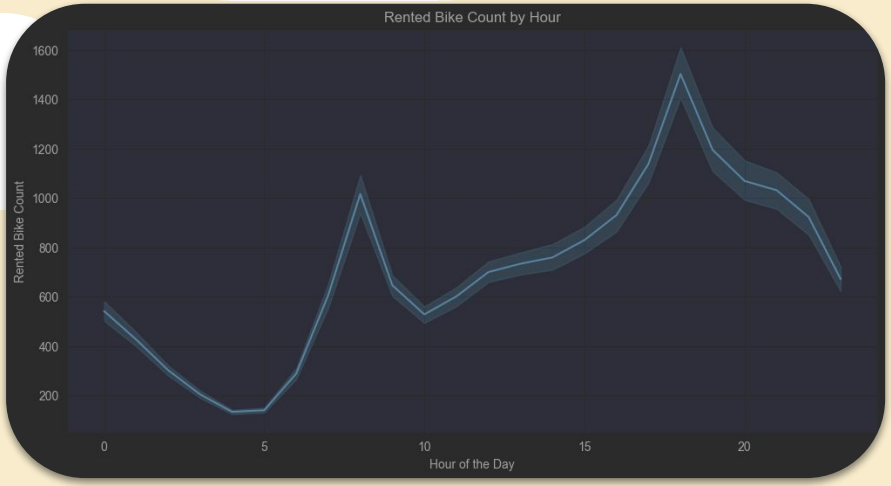
DATA PRE-PROCESSING / 2

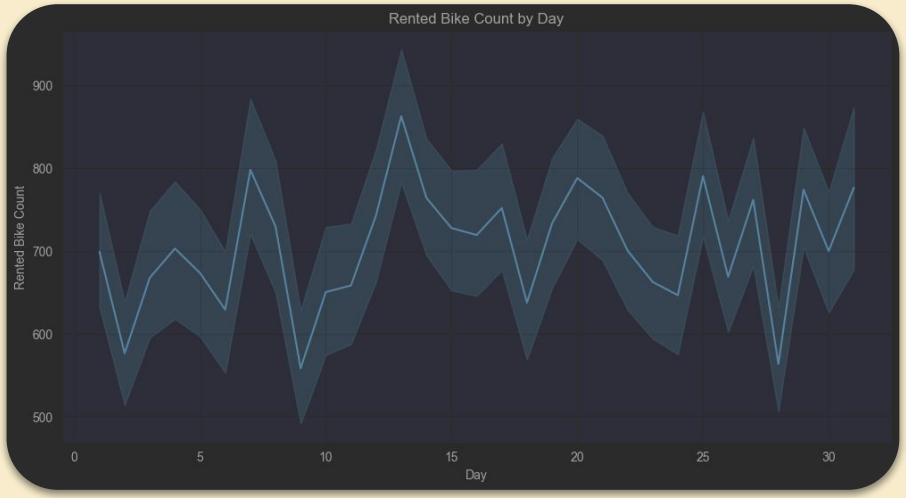
```
df['Day'] = df['Date'].dt.day
df['Month'] = df['Date'].dt.month
df['Year'] = df['Date'].dt.year
df['WeekDay']=df['Date'].dt.day_name()
df.drop(columns=['Date'], inplace=True)
```

Analyze the data by day and month
Correlated values

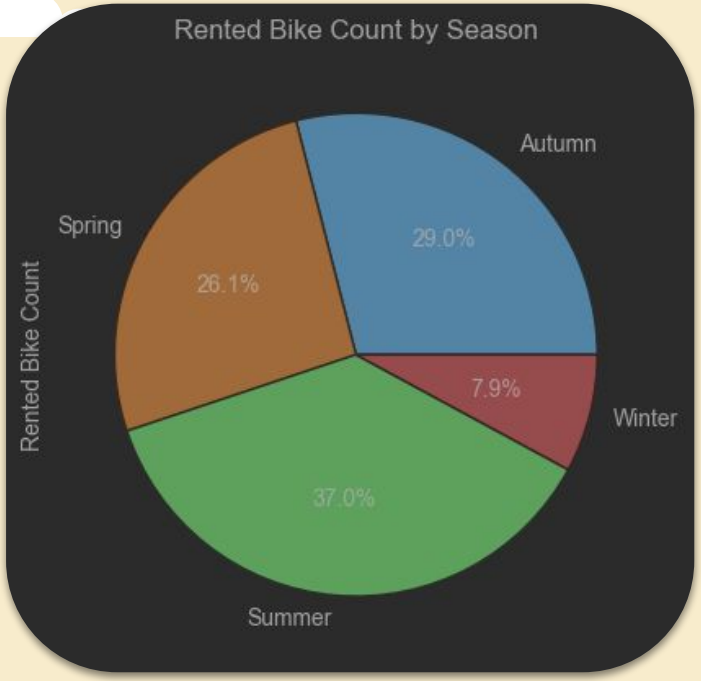


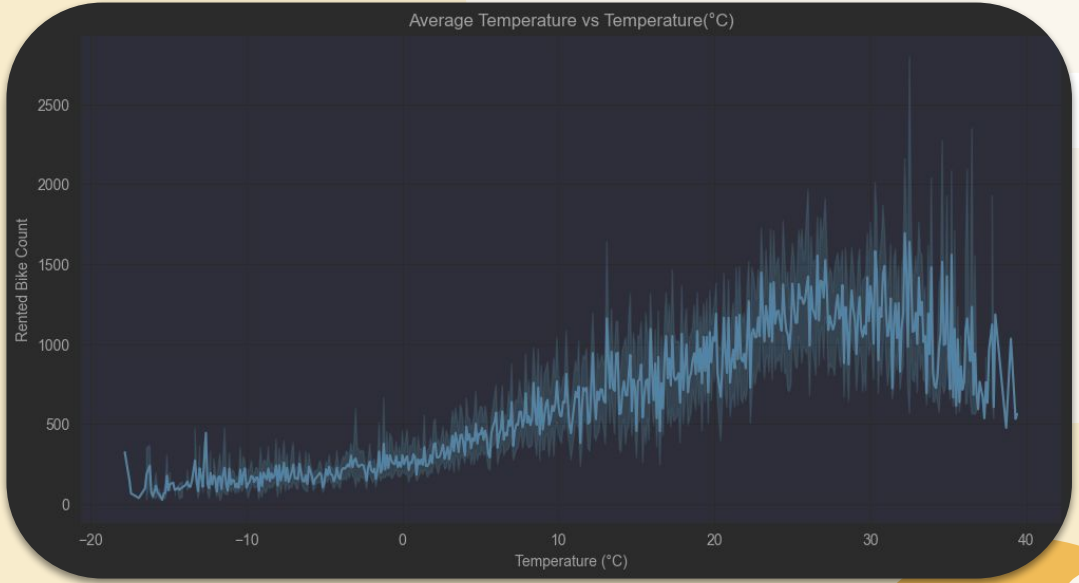
DATA VISUALIZATION / 1





DATA VISUALIZATION / 2





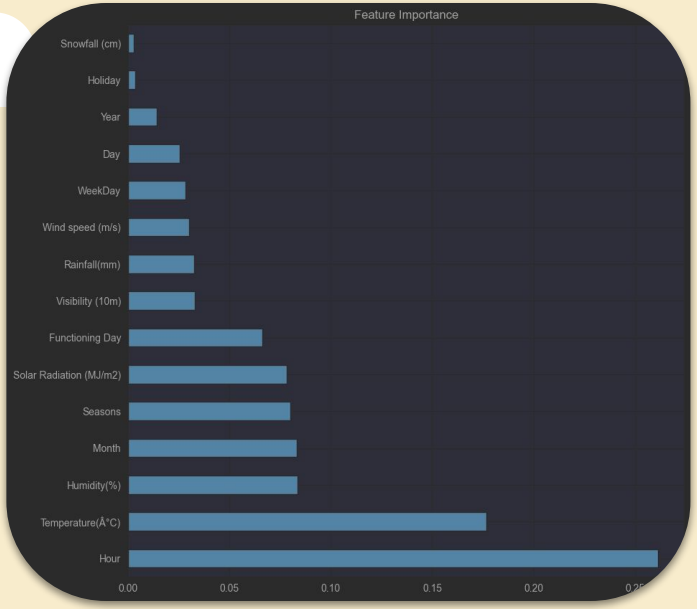
MODELING / 1

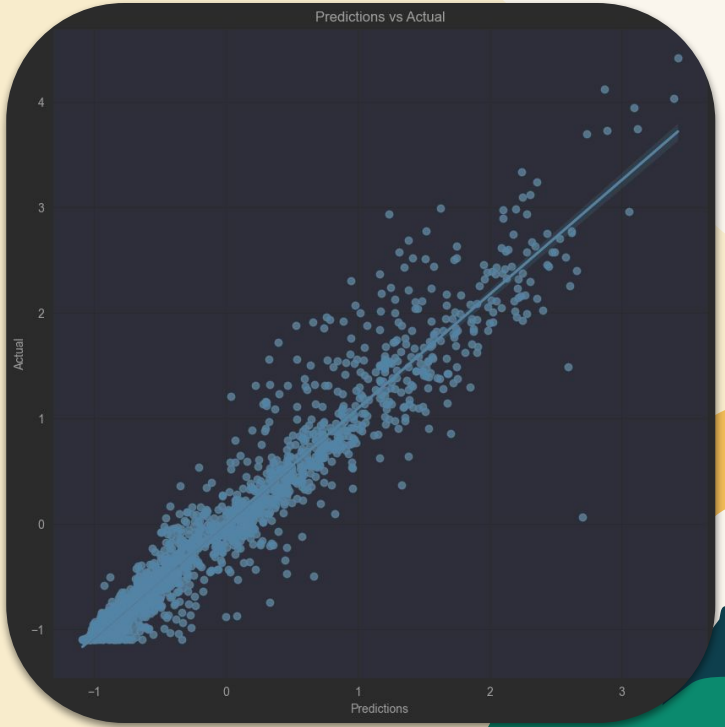
Model	Adjusted R-Squared	R-Squared	RMSE	Time Taken
HistGradientBoostingRegressor	0.94	0.94	0.24	0.49
LGBMRegressor	0.94	0.94	0.24	0.11
ExtraTreesRegressor	0.94	0.94	0.24	4.07
XGBRegressor	0.94	0.94	0.25	0.22
RandomForestRegressor	0.93	0.93	0.27	8.01

```
Best parameters: {'bootstrap': False, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 200}
```

```
Mean Squared Error: 0.08255609946587276
Mean Absolute Error: 0.1740232928362238
R^2 Score: 0.9175770526251155
```

MODELING / 2


Feature Importance
Snowfall (cm)
Holiday
Year
Day
WeekDay
Wind speed (m/s)
Rainfall(mm)
Visibility (10m)
Functioning Day
Solar Radiation (MJ/m2)
Seasons
Month
Humidity(%)
Temperature(Â°C)
Hour
0.00
0.05
0.10
0.15
0.20


Predictions vs Actual
Actual
Predictions
4
3
2
1
0
−1
−1
0
1
2
3

04

API

How did we transformed our model into an API?

PREDICTION

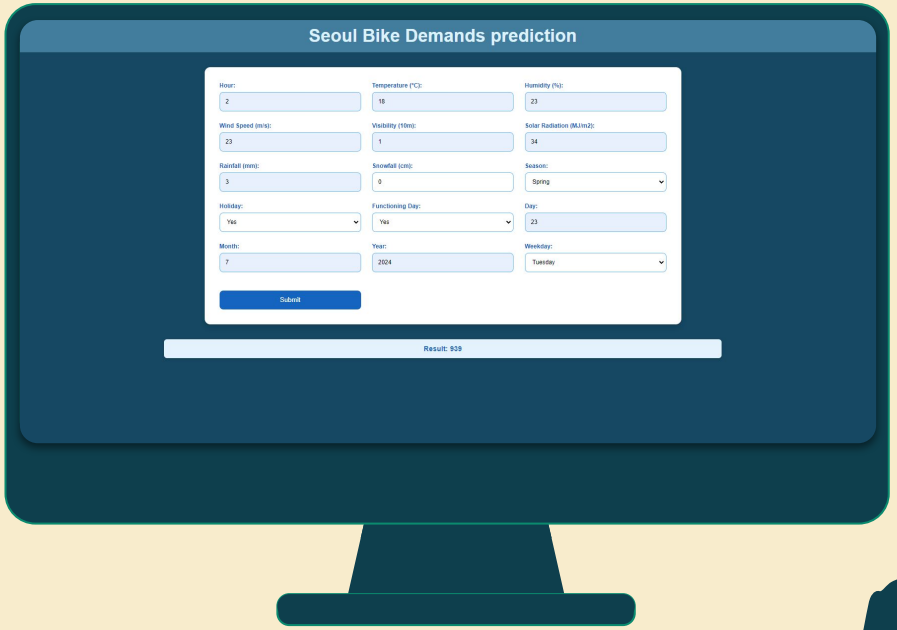
Pickle to read our model and use it

FAST API

Ensuring efficient handling of requests and responses

REACT

Familiar with it. Create a form where the user inserts his data



THANKS!

Do you have any questions?