RLM_Final

November 29, 2025

importation des bibliotheques

[11]:

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import LabelEncoder
from sklearn.preprocessing import StandardScaler
from sklearn.experimental import enable_iterative_imputer
from sklearn.impute import IterativeImputer
```

[12]:

```
data = "taxi_trip_pricing.csv"

df = pd.read_csv(data)
```

[13]:

```
x1 = df.select_dtypes(include = [np.number])
x2 = df.select_dtypes(exclude = [np.number])
```

[14]:

```
impute = IterativeImputer()

x1 = pd.DataFrame(impute.fit_transform(x1),columns = x1.columns)
```

[15]:

```
encode = LabelEncoder()

for col in x2.columns :
    x2[col] = x2[col].fillna(x2[col].mode()[0])
    x2[col] = encode.fit_transform(x2[col])
```

[16]:

```
x = pd.DataFrame(pd.concat([x1,x2],axis = 1))
```

[17]:

```
def outlier(col):
    Q1 = col.quantile(0.25)
    Q3 = col.quantile(0.75)

    IQR = Q3 - Q1

    Fb = Q1 - 1.5 * IQR
    Fh = Q3 + 1.5 * IQR

    return col.clip(lower = Fb, upper = Fh)
```

```
for col in x.columns :
    x[col] = outlier(x[col])
```

[18]:

```
y = x['Trip_Price']
```

[19]:

```
x = x.drop(columns = 'Trip_Price')
```

[20]:

```
scale = StandardScaler()

x_scaled = pd.DataFrame(scale.fit_transform(x),columns = x.columns)
```

[21]:

```
X = np.hstack((x_scaled,np.ones((x_scaled.shape[0],1))))
```

[22]:

```
Y = y.values.reshape(-1,1)
```

importation de la class RLM(regression lineaire manuel)

[23]:

```
from models_ai.RLM import RLM
```

[24]:

```
model = RLM(lr = 0.01,ite = 1000,scaler = scale)
```

[25]:

```
model.fit(X,Y)
```

Extraction du modele entainé

[ ]:

```
import joblib

joblib.dump(model, "RLM.pkl")
```