

Master 2 Business Intelligence

Projet performance décisionnelle :

Développement d'un prédicteur de délai pour prédire en temps réel le
temps d'attente des clients

Présenté et soutenu par :
Abdoulaye DIANKHA

SOMMAIRE

	INTRODUCTION	1
1	CHAPITRE 1 : Découpage des données du mois jours par jours et conception du jeux de données.....	3

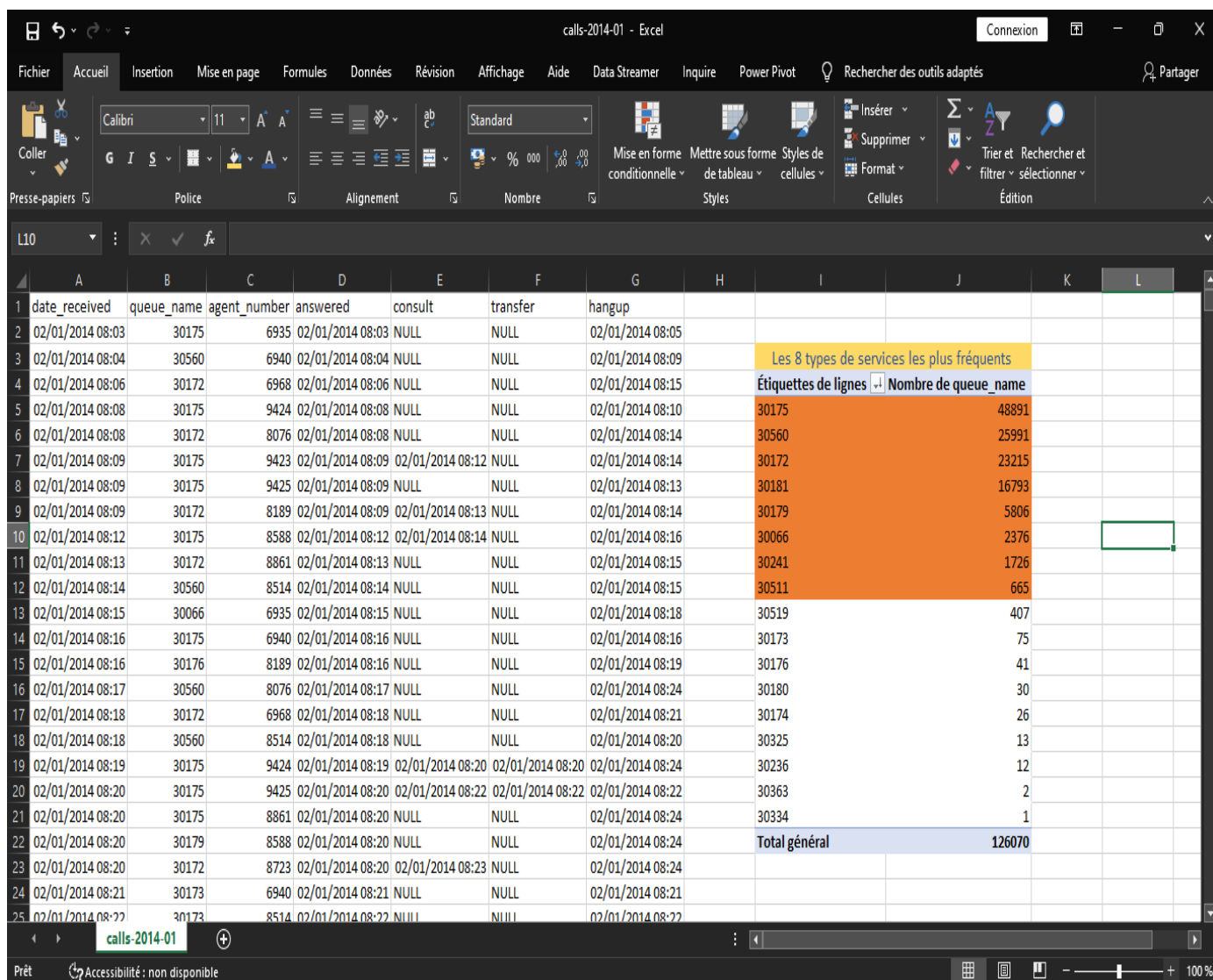
.....

INTRODUCTION GENERALE

Dans ce projet, nous commençons par l'étude de la prédiction de délai d'attente des clients dans les centres d'appels multi-compétences. Le temps d'attente a un impact important sur la qualité du service perçue par les clients. L'annonce du délai d'attente permet de réduire l'incertitude du client à propos de son délai d'attente. Elle peut également augmenter la satisfaction du client et réduire le nombre d'abandons. Ceci nécessite d'avoir un bon prédicteur de délai. Malheureusement, les prédicteurs existants ne sont pas adaptés pour les centres d'appels multi-compétence.

1 CHAPITRE 1 : Découpage des données du mois jours par jours et conception du jeu de données

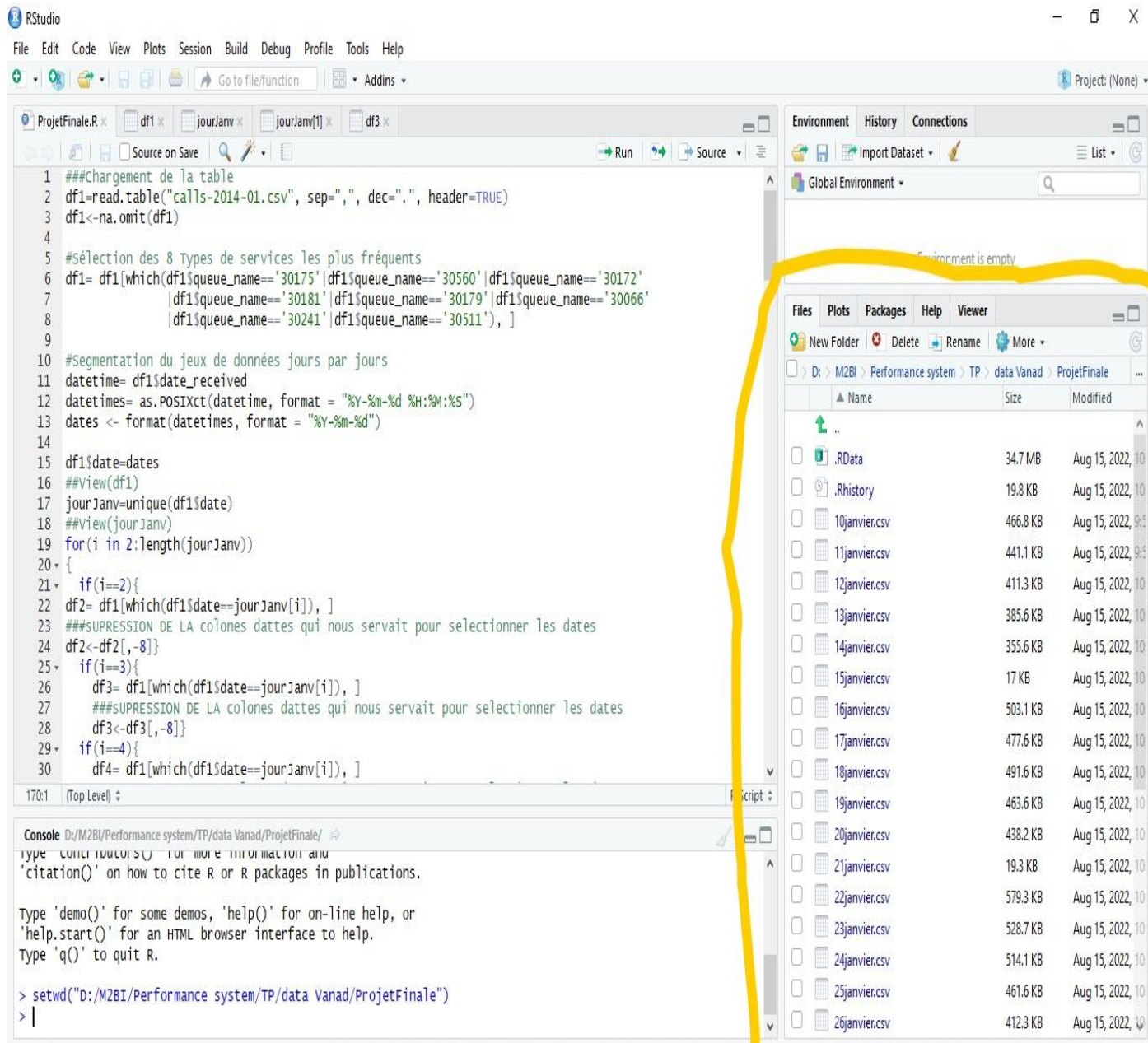
1.1 Les 8 types de services les plus fréquents



date_received	queue_name	agent_number	answered	consult	transfer	hangup
02/01/2014 08:03	30175	6935	02/01/2014 08:03	NULL	NULL	02/01/2014 08:05
02/01/2014 08:04	30560	6940	02/01/2014 08:04	NULL	NULL	02/01/2014 08:09
02/01/2014 08:06	30172	6968	02/01/2014 08:06	NULL	NULL	02/01/2014 08:15
02/01/2014 08:08	30175	9424	02/01/2014 08:08	NULL	NULL	02/01/2014 08:10
02/01/2014 08:08	30172	8076	02/01/2014 08:08	NULL	NULL	02/01/2014 08:14
02/01/2014 08:09	30175	9423	02/01/2014 08:09	02/01/2014 08:12	NULL	02/01/2014 08:14
02/01/2014 08:09	30175	9425	02/01/2014 08:09	NULL	NULL	02/01/2014 08:13
02/01/2014 08:09	30172	8189	02/01/2014 08:09	02/01/2014 08:13	NULL	02/01/2014 08:14
02/01/2014 08:12	30175	8588	02/01/2014 08:12	02/01/2014 08:14	NULL	02/01/2014 08:16
02/01/2014 08:13	30172	8861	02/01/2014 08:13	NULL	NULL	02/01/2014 08:15
02/01/2014 08:14	30560	8514	02/01/2014 08:14	NULL	NULL	02/01/2014 08:15
02/01/2014 08:15	30066	6935	02/01/2014 08:15	NULL	NULL	02/01/2014 08:18
02/01/2014 08:16	30175	6940	02/01/2014 08:16	NULL	NULL	02/01/2014 08:16
02/01/2014 08:16	30176	8189	02/01/2014 08:16	NULL	NULL	02/01/2014 08:19
02/01/2014 08:17	30560	8076	02/01/2014 08:17	NULL	NULL	02/01/2014 08:24
02/01/2014 08:18	30172	6968	02/01/2014 08:18	NULL	NULL	02/01/2014 08:21
02/01/2014 08:18	30560	8514	02/01/2014 08:18	NULL	NULL	02/01/2014 08:20
02/01/2014 08:19	30175	9424	02/01/2014 08:19	02/01/2014 08:20	02/01/2014 08:20	02/01/2014 08:24
02/01/2014 08:20	30175	9425	02/01/2014 08:20	02/01/2014 08:22	02/01/2014 08:22	02/01/2014 08:22
02/01/2014 08:20	30175	8861	02/01/2014 08:20	NULL	NULL	02/01/2014 08:24
02/01/2014 08:20	30179	8588	02/01/2014 08:20	NULL	NULL	02/01/2014 08:24
02/01/2014 08:20	30172	8723	02/01/2014 08:20	02/01/2014 08:23	NULL	02/01/2014 08:24
02/01/2014 08:21	30173	6940	02/01/2014 08:21	NULL	NULL	02/01/2014 08:21
02/01/2014 08:22	30173	8514	02/01/2014 08:22	NULL	NULL	02/01/2014 08:22

Les 8 types de services les plus fréquents	Étiquettes de lignes	Nombre de queue_name
30175		48891
30560		25991
30172		23215
30181		16793
30179		5806
30066		2376
30241		1726
30511		665
30519		407
30173		75
30176		41
30180		30
30174		26
30325		13
30236		12
30363		2
30334		1
Total général		126070

1.2 Code R pour le découpage de données



The screenshot displays the RStudio interface with a script editor, environment pane, and file explorer.

Script Editor:

```
1 ##Chargement de la table
2 df1=read.table("calls-2014-01.csv", sep=";", dec=".", header=TRUE)
3 df1<-na.omit(df1)
4
5 #sélection des 8 Types de services les plus fréquents
6 df1= df1[which(df1$queue_name=='30175'|df1$queue_name=='30560'|df1$queue_name=='30172'
7             |df1$queue_name=='30181'|df1$queue_name=='30179'|df1$queue_name=='30066'
8             |df1$queue_name=='30241'|df1$queue_name=='30511'), ]
9
10 #Segmentation du jeux de données jours par jours
11 datetime= df1$date_received
12 datetimes= as.POSIXct(datetime, format = "%Y-%m-%d %H:%M:%S")
13 dates <- format(datetimes, format = "%Y-%m-%d")
14
15 df1$date=dates
16 ##View(df1)
17 jourJanv=unique(df1$date)
18 ##View(jourJanv)
19 for(i in 2:length(jourJanv))
20 {
21   if(i==2){
22     df2= df1[which(df1$date==jourJanv[i]), ]
23     ##SUPPRESSION DE LA colonnes dattes qui nous servait pour selectionner les dates
24     df2<-df2[,-8]}
25   if(i==3){
26     df3= df1[which(df1$date==jourJanv[i]), ]
27     ##SUPPRESSION DE LA colonnes dattes qui nous servait pour selectionner les dates
28     df3<-df3[,-8]}
29   if(i==4){
30     df4= df1[which(df1$date==jourJanv[i]), ]
```

Environment Pane: Shows "Global Environment" with "Environment is empty".

File Explorer: Displays the file structure of the project. The path is `D:\M2BI\Performance system\TP\data Vanad\ProjetFinale`. The files listed are:

Name	Size	Modified
..		
.RData	34.7 MB	Aug 15, 2022, 10:00
.Rhistory	19.8 KB	Aug 15, 2022, 10:00
10janvier.csv	466.8 KB	Aug 15, 2022, 9:55
11janvier.csv	441.1 KB	Aug 15, 2022, 9:55
12janvier.csv	411.3 KB	Aug 15, 2022, 10:00
13janvier.csv	385.6 KB	Aug 15, 2022, 10:00
14janvier.csv	355.6 KB	Aug 15, 2022, 10:00
15janvier.csv	17 KB	Aug 15, 2022, 10:00
16janvier.csv	503.1 KB	Aug 15, 2022, 10:00
17janvier.csv	477.6 KB	Aug 15, 2022, 10:00
18janvier.csv	491.6 KB	Aug 15, 2022, 10:00
19janvier.csv	463.6 KB	Aug 15, 2022, 10:00
20janvier.csv	438.2 KB	Aug 15, 2022, 10:00
21janvier.csv	19.3 KB	Aug 15, 2022, 10:00
22janvier.csv	579.3 KB	Aug 15, 2022, 10:00
23janvier.csv	528.7 KB	Aug 15, 2022, 10:00
24janvier.csv	514.1 KB	Aug 15, 2022, 10:00
25janvier.csv	461.6 KB	Aug 15, 2022, 10:00
26janvier.csv	412.3 KB	Aug 15, 2022, 10:00

Console: Shows the current working directory set to `D:/M2BI/Performance system/TP/data Vanad/ProjetFinale/`.

1.2.1 Code java de simulation et de création de jeux de données pour le développement d'un prédicteur de délai pour prédire en temps réel le temps d'attente des clients

```

187
188
189 }
190
191 public void createDataset() throws IOException{
192     String x=null;
193     String y=null;
194     List<String> fichier = Arrays.asList("data2janvier.csv","data3janvier.csv","data4janvier.csv","data5janvier.csv","data6janvier.csv","data7janvier.csv",
195     "data8janvier.csv","data9janvier.csv","data10janvier.csv","data11janvier.csv","data12janvier.csv","data13janvier.csv","data14janvier.csv","data15janvier.csv",
196     "data16janvier.csv","data17janvier.csv","data18janvier.csv","data19janvier.csv","data20janvier.csv","data21janvier.csv","data22janvier.csv","data23janvier.csv",
197     "data24janvier.csv","data25janvier.csv","data26janvier.csv");
198     for(String fichiers: fichier){
199         BufferedWriter br=new BufferedWriter(new FileWriter(fichiers));
200         String entete="Type;"+LE5;"+NAgent;"+Lf1;
201         "+Lf2;"+Lf3;"+Lf4;"+Lf5;"+Lf6;"+Lf7;"+Lf8;"+WaitingTime;
202         br.write(entete+"\n");
203         for(Customer c:EndServiceCust)
204             { y="c.waitingTime;"+c.arrivalTime;"+c.LE5;"+c.numberAgents;"+c.lf1;"+c.lf2;"+c.lf3;"+c.lf4;"+c.lf5;"+
205             "+c.lf6;"+c.lf7;"+c.lf8;";
206             br.write(x+";"+y+"\n");
207         }
208     }
209     br.close();
210 }
211
212 class EndOfSim extends Event{
213     public void actions(){
214         Sim.stop();
215     }
216 }
217
218 public void simulateOneDay( String file) throws IOException{
219     Sim.init();
220     readFile(file);
221 }

```

Markers Properties Servers Data Source Explorer Snippets Console Coverage

<terminated> TravaillFinal [Java Application] C:\Users\HP\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_18.0.1.v20220515-1614\jre\bin\javaw.exe (15 août 2022, 23:20:43 - 23:20:44)

02/01/2014 19:13;30172;7479;02/01/2014 19:13;NULL;NULL;02/01/2014 19:18
02/01/2014 19:13;30560;6971;02/01/2014 19:14;NULL;NULL;02/01/2014 19:16
02/01/2014 19:22;30560;6971;02/01/2014 19:22;NULL;NULL;02/01/2014 19:28
02/01/2014 19:23;30175;7479;02/01/2014 19:23;NULL;NULL;02/01/2014 19:30
02/01/2014 19:25;30175;8861;02/01/2014 19:25;NULL;NULL;02/01/2014 19:26

```

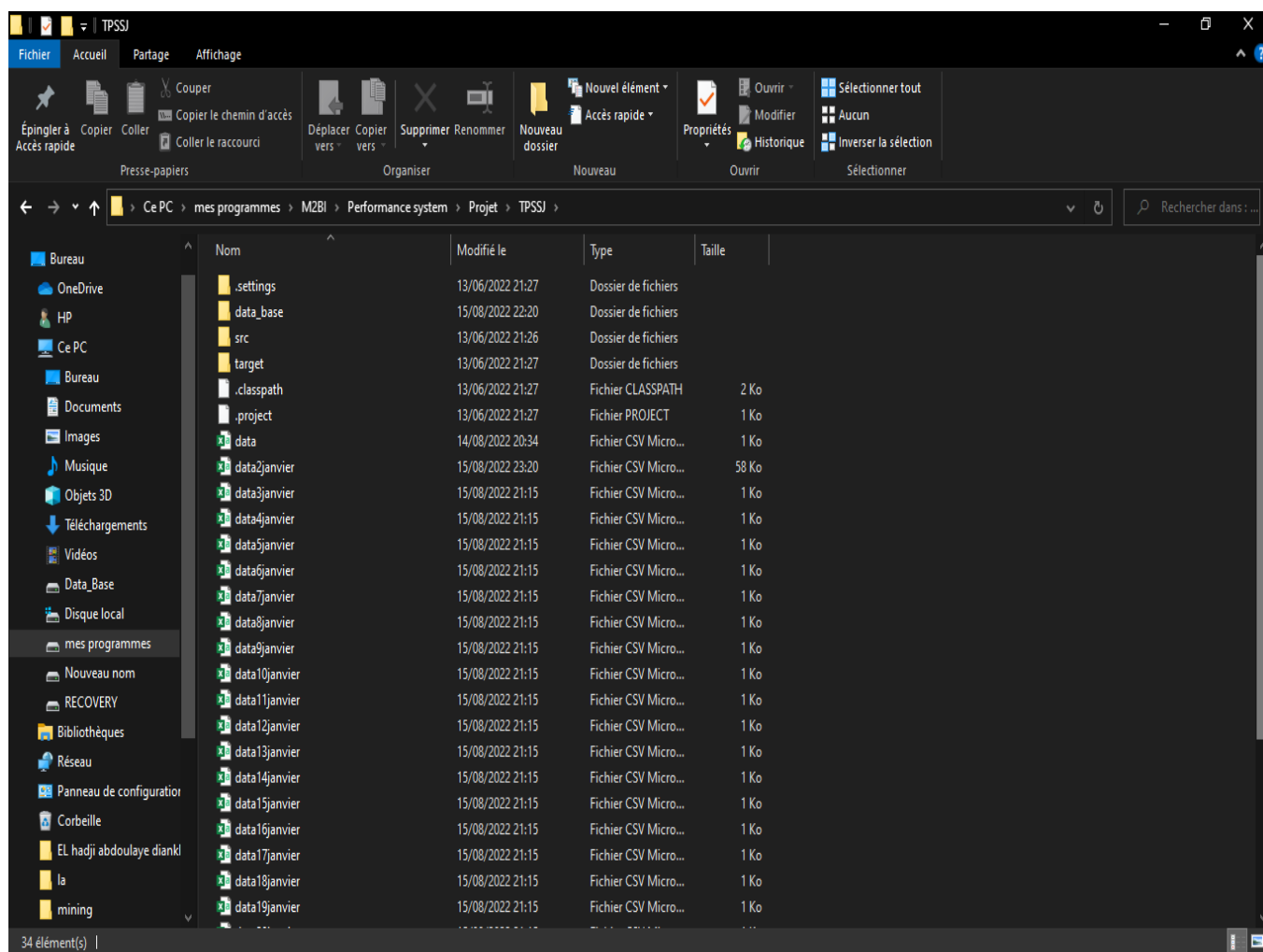
205     x=c.type+";"+c.arrivalTime;"+c.LE5;"+c.numberAgents;"+c.lf1;"+c.lf2;"+c.lf3;"+c.lf4;"+c.lf5;"+
206     "+c.lf6;"+c.lf7;"+c.lf8;";
207     br.write(x+";"+y+"\n");
208 }
209 br.close();
210 }
211
212 class EndOfSim extends Event{
213     public void actions(){
214         Sim.stop();
215     }
216 }
217
218 public void simulateOneDay( String file) throws IOException{
219     Sim.init();
220     readFile(file);
221     new EndOfSim().schedule (43200);
222     Sim.start();
223 }
224
225 public static void main(String args[]) throws IOException{
226     TravaillFinal tf=new TravaillFinal();
227     List<String> fichier = Arrays.asList("2janvier.csv","3janvier.csv","4janvier.csv","5janvier.csv","6janvier.csv","7janvier.csv",
228     "8janvier.csv","9janvier.csv","10janvier.csv","11janvier.csv","12janvier.csv","13janvier.csv","14janvier.csv","15janvier.csv","16janvier.csv",
229     "17janvier.csv","18janvier.csv","19janvier.csv","20janvier.csv","21janvier.csv","22janvier.csv","23janvier.csv","24janvier.csv",
230     "25janvier.csv","26janvier.csv");
231     for(String line: fichier) {
232         tf.simulateOneDay("data_base/"+line);
233         tf.createDataset();
234     }
235 }
236 }

```

Markers Properties Servers Data Source Explorer Snippets Console Coverage

<terminated> TravaillFinal [Java Application] C:\Users\HP\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_18.0.1.v20220515-1614\jre\bin\javaw.exe (15 août 2022, 23:20:43 - 23:20:44)

02/01/2014 19:13;30172;7479;02/01/2014 19:13;NULL;NULL;02/01/2014 19:18
02/01/2014 19:13;30560;6971;02/01/2014 19:14;NULL;NULL;02/01/2014 19:16
02/01/2014 19:22;30560;6971;02/01/2014 19:22;NULL;NULL;02/01/2014 19:28
02/01/2014 19:23;30175;7479;02/01/2014 19:23;NULL;NULL;02/01/2014 19:30
02/01/2014 19:25;30175;8861;02/01/2014 19:25;NULL;NULL;02/01/2014 19:26



1.2.2 Prenons l'exemple d'ouvrir le fichier data2janvier.csv pour voir le résultat obtenu après simulation

Type	ArrivalTime	LES	NAgent	Lf1	Lf2	Lf3	Lf4	Lf5	Lf6	Lf7	Lf8	WaitingTime
30175	28980.0	0.0	0	1	0	0	0	0	0	0	0	120.0
30560	29040.0	0.0	1	0	1	0	0	0	0	0	0	300.0
30175	29280.0	120.0	2	1	0	0	0	0	0	0	0	120.0
30175	29340.0	120.0	4	1	0	0	0	0	0	0	0	240.0
30172	29280.0	540.0	2	0	0	1	0	0	0	0	0	360.0
30175	29340.0	120.0	4	1	0	0	0	0	0	0	0	300.0
30172	29340.0	360.0	4	0	0	1	0	0	0	0	0	300.0
30172	29160.0	0.0	1	0	0	1	0	0	0	0	0	540.0
30172	29580.0	300.0	6	0	0	1	0	0	0	0	0	120.0
30560	29640.0	300.0	6	0	1	0	0	0	0	0	0	60.0
30175	29520.0	240.0	5	1	0	0	0	0	0	0	0	240.0
30175	29760.0	240.0	2	1	0	0	0	0	0	0	0	0.0
30066	29700.0	0.0	4	0	0	0	0	0	1	0	0	180.0
30560	29880.0	420.0	2	0	1	0	0	0	0	0	0	120.0
30172	29880.0	120.0	2	0	0	1	0	0	0	0	0	180.0
30175	30000.0	300.0	4	1	0	0	0	0	0	0	0	120.0
30560	29820.0	60.0	1	0	1	0	0	0	0	0	0	420.0
30175	29940.0	0.0	3	1	0	0	0	0	0	0	0	300.0
30175	30000.0	300.0	4	1	0	0	0	0	0	0	0	240.0
30179	30000.0	0.0	4	0	0	0	0	1	0	0	0	240.0
30172	30000.0	180.0	4	0	0	1	0	0	0	0	0	240.0
30179	30180.0	480.0	6	0	0	0	0	1	0	0	0	60.0
30179	30240.0	60.0	8	0	0	0	0	1	0	0	0	180.0
30175	30180.0	240.0	6	1	0	0	0	0	0	0	0	360.0

1.3 Explication du fichier data2janvier.csv

- Dans ce jeu de données ci-dessus j'ai considéré que :
- queue_name=30175=lf1 , queue_name=30560=lf2 , queue_name=30172=lf3,
queue_name=30181=lf4,queue_name=30179=lf5,queue_name=30066=lf6,
queue_name=30241=lf7,queue_name=30511=lf8

1.4 Explication des lf : (prenons l'exemple de lf1

Si lf1=1 cela veut dire que la queue_name=30175 est entrée en service et tous les autres lf qui sont égale à 0 sans la file d'attente.

1.5 CONSTRUCTION DU MODELE DE PREDICTION

Jupyter prédire en temps réel du temps d'attente des clients Dernière Sauvegarde : il y a une heure (auto-sauvegardé)

File Edit View Insert Cell Kernel Widgets Help

Entrée [101]: `import pandas as pd
data=pd.read_csv("data2janvier.csv",sep=";")
data`

Out[101]:

	Type	ArrivalTime	LES	NAgent	Lf1	Lf2	Lf3	Lf4	Lf5	Lf6	Lf7	Lf8	WaitingTime
0	30175	28980.0	0.0	0	1	0	0	0	0	0	0	0	120.0
1	30560	29040.0	0.0	1	0	1	0	0	0	0	0	0	300.0
2	30175	29280.0	120.0	2	1	0	0	0	0	0	0	0	120.0
3	30175	29340.0	120.0	4	1	0	0	0	0	0	0	0	240.0
4	30172	29280.0	540.0	2	0	0	1	0	0	0	0	0	360.0
...
1320	30179	42780.0	360.0	58	0	0	0	0	1	0	0	0	360.0
1321	30066	42780.0	60.0	58	0	0	0	0	0	1	0	0	360.0
1322	30179	42840.0	360.0	69	0	0	0	0	1	0	0	0	300.0
1323	30560	43080.0	240.0	77	0	1	0	0	0	0	0	0	60.0
1324	30560	43080.0	240.0	77	0	1	0	0	0	0	0	0	60.0

1325 rows x 13 columns

Entrée [102]: `data=data.drop(["ArrivalTime","LES","NAgent","Lf1","Lf2","Lf3","Lf4","Lf5","Lf6","Lf7","Lf8"], axis=1)`

Entrée [103]: `data`

Out[103]:

Jupyter prédire en temps réel du temps d'attente des clients Dernière Sauvegarde : il y a une heure (auto-sauvegardé)

File Edit View Insert Cell Kernel Widgets Help

Entrée [102]: `data=data.drop(["ArrivalTime","LES","NAgent","Lf1","Lf2","Lf3","Lf4","Lf5","Lf6","Lf7","Lf8"], axis=1)`

Entrée [103]: `data`

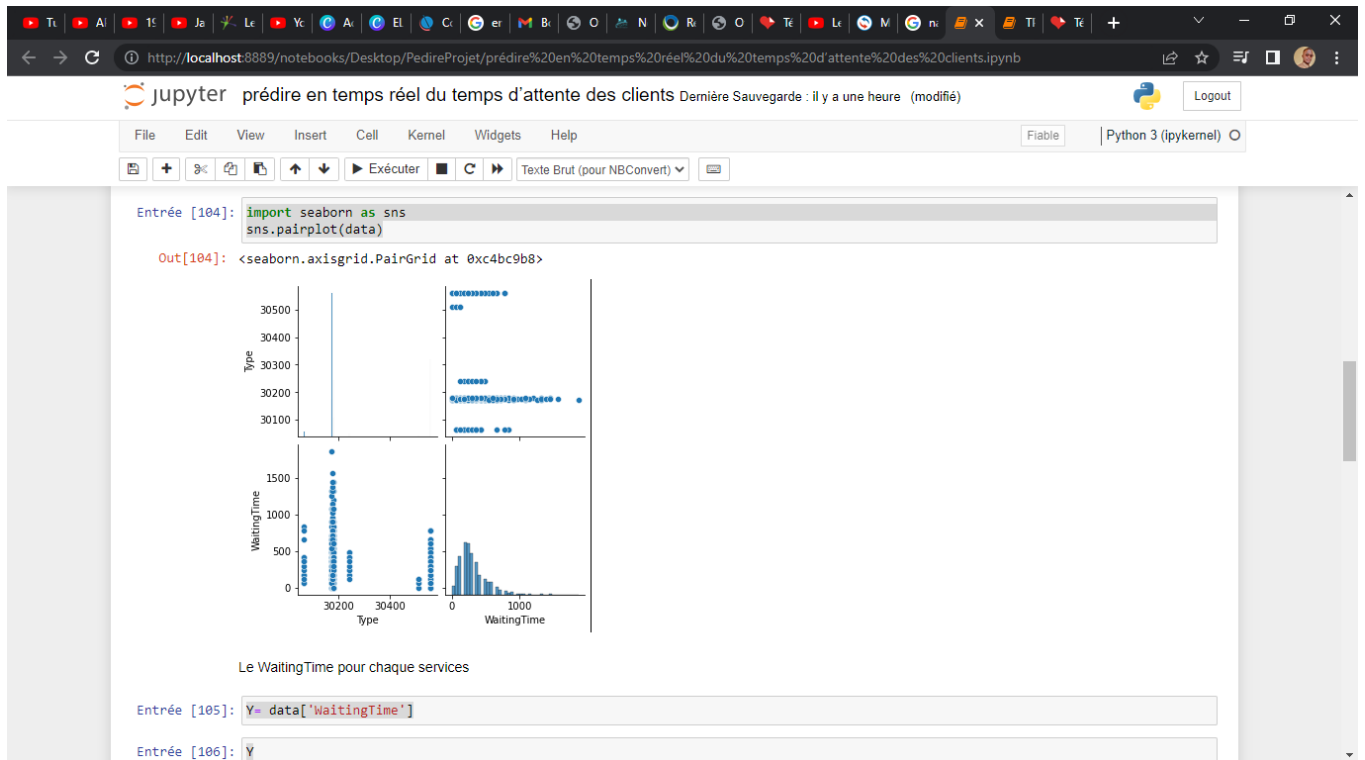
Out[103]:

	Type	WaitingTime
0	30175	120.0
1	30560	300.0
2	30175	120.0
3	30175	240.0
4	30172	360.0
...
1320	30179	360.0
1321	30066	360.0
1322	30179	300.0
1323	30560	60.0
1324	30560	60.0

1325 rows x 2 columns

Entrée [104]: `import seaborn as sns
sns.pairplot(data)`

Out[104]: `<seaborn.axisgrid.PairGrid at 0xc4bc9b8>`



Jupyter prédire en temps réel du temps d'attente des clients Dernière Sauvegarde : il y a une heure (modifié)

File Edit View Insert Cell Kernel Widgets Help Python 3 (ipykernel)

```
Entrée [105]: Y = data['WaitingTime']
```

```
Entrée [106]: Y
```

Out[106]:

0	120.0
1	300.0
2	120.0
3	240.0
4	360.0
...	
1320	360.0
1321	360.0
1322	300.0
1323	60.0
1324	60.0

Name: WaitingTime, Length: 1325, dtype: float64

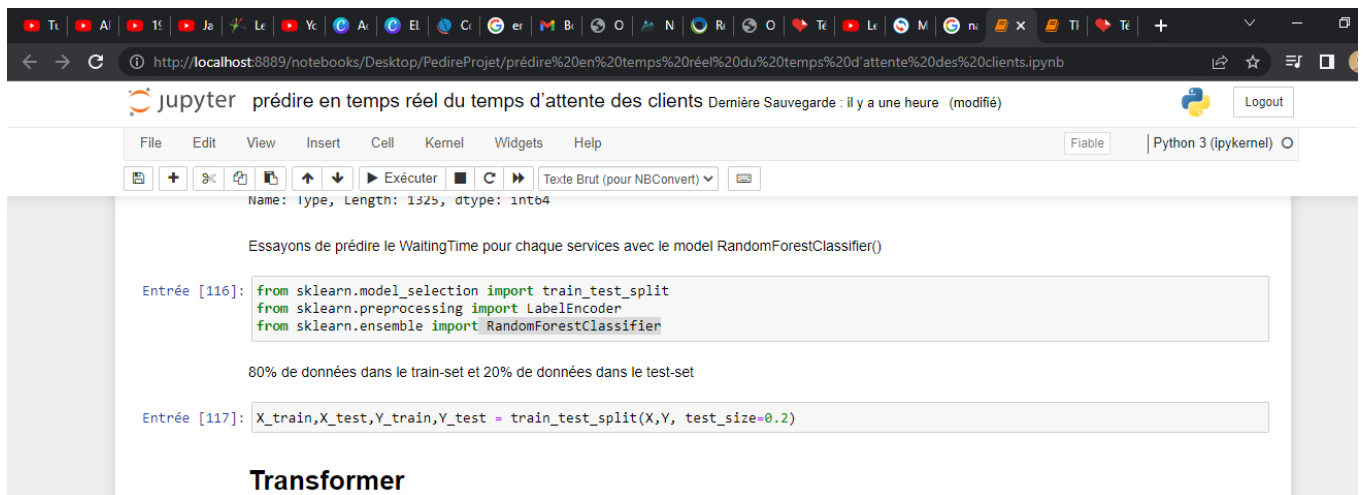
Les différents types de services

```
Entrée [107]: X = data['Type']
```

```
Entrée [110]: X
```

Out[110]:

0	30175
1	30560
2	30175
3	30175
4	30172
...	
1320	30179
1321	30066



http://localhost:8889/notebooks/Desktop/PedireProjet/prédire%20en%20temps%20réel%20du%20temps%20d'attente%20des%20clients.ipynb

jupyter prédire en temps réel du temps d'attente des clients Dernière Sauvegarde : il y a une heure (modifié)

File Edit View Insert Cell Kernel Widgets Help

Exécuter

Name: lype, Length: 1325, dtype: int64

Essays de prédire le WaitingTime pour chaque services avec le model RandomForestClassifier()

Entrée [116]:

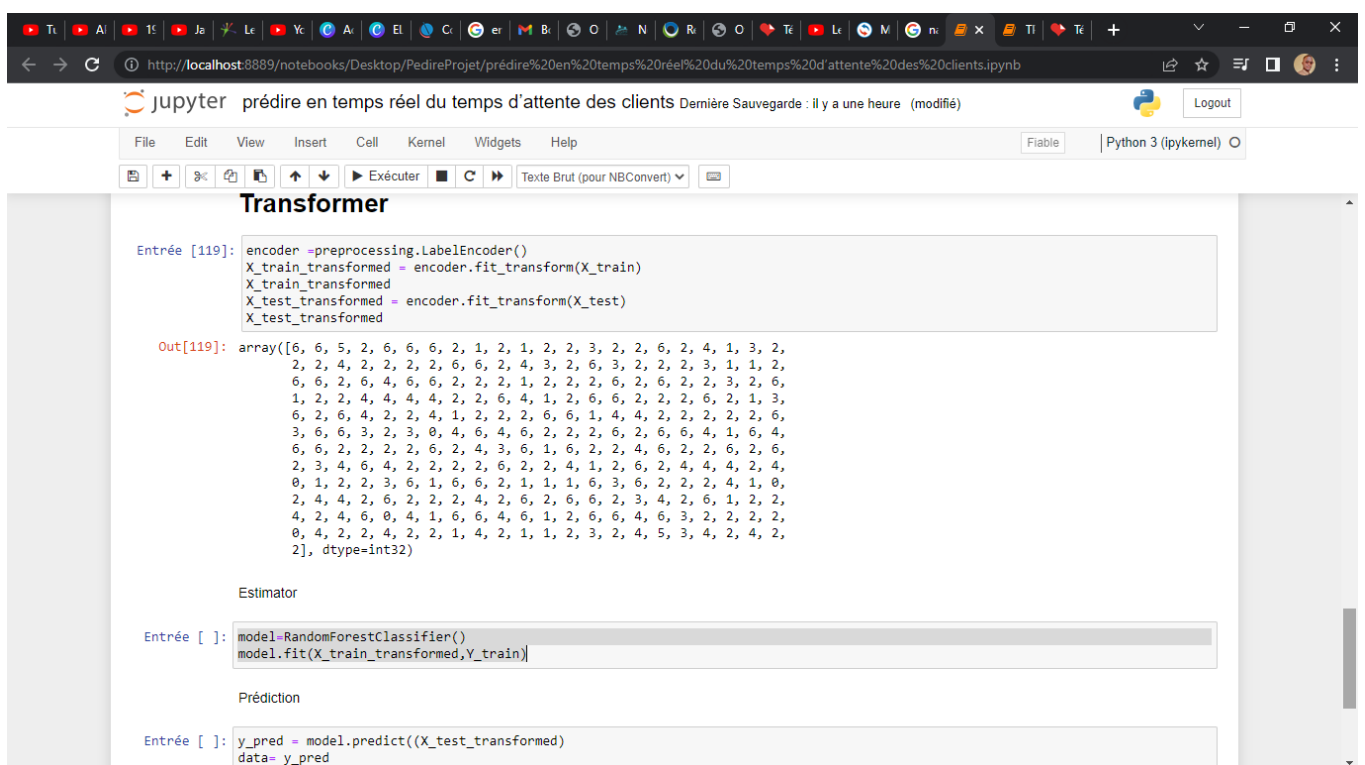
```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.ensemble import RandomForestClassifier
```

80% de données dans le train-set et 20% de données dans le test-set

Entrée [117]:

```
X_train,X_test,Y_train,Y_test = train_test_split(X,Y, test_size=0.2)
```

Transformer



http://localhost:8889/notebooks/Desktop/PedireProjet/prédire%20en%20temps%20réel%20du%20temps%20d'attente%20des%20clients.ipynb

jupyter prédire en temps réel du temps d'attente des clients Dernière Sauvegarde : il y a une heure (modifié)

File Edit View Insert Cell Kernel Widgets Help

Exécuter

Transformer

Entrée [119]:

```
encoder = preprocessing.LabelEncoder()
X_train_transformed = encoder.fit_transform(X_train)
X_train_transformed
X_test_transformed = encoder.fit_transform(X_test)
X_test_transformed
```

Out[119]:

```
array([[6, 6, 5, 2, 6, 6, 6, 2, 1, 2, 1, 2, 2, 3, 2, 2, 6, 2, 4, 1, 3, 2,
        2, 2, 4, 2, 2, 2, 6, 6, 2, 4, 3, 2, 6, 3, 2, 2, 2, 3, 1, 1, 2,
        6, 6, 2, 6, 4, 6, 6, 2, 2, 2, 1, 2, 2, 2, 6, 2, 6, 2, 2, 3, 2, 6,
        1, 2, 2, 4, 4, 4, 4, 2, 2, 6, 4, 1, 2, 6, 6, 2, 2, 2, 6, 2, 1, 3,
        6, 2, 6, 4, 2, 2, 4, 1, 2, 2, 2, 6, 6, 1, 4, 4, 2, 2, 2, 2, 2, 6,
        3, 6, 6, 3, 2, 3, 0, 4, 6, 4, 6, 2, 2, 2, 6, 2, 6, 6, 4, 1, 6, 4,
        6, 6, 2, 2, 2, 2, 6, 2, 4, 3, 6, 1, 6, 2, 2, 4, 6, 2, 2, 6, 2, 6,
        2, 3, 4, 6, 4, 2, 2, 2, 6, 2, 2, 4, 1, 2, 6, 2, 4, 4, 4, 2, 4,
        0, 1, 2, 2, 3, 6, 1, 6, 6, 2, 1, 1, 1, 6, 3, 6, 2, 2, 2, 4, 1, 0,
        2, 4, 4, 2, 6, 2, 2, 2, 4, 2, 6, 2, 6, 6, 2, 3, 4, 2, 6, 1, 2, 2,
        4, 2, 4, 6, 0, 4, 1, 6, 6, 4, 6, 1, 2, 6, 6, 4, 6, 3, 2, 2, 2,
        0, 4, 2, 2, 4, 2, 2, 1, 4, 2, 1, 1, 2, 3, 2, 4, 5, 3, 4, 2, 4, 2,
        2], dtype=int32)
```

Estimator

Entrée []:

```
model=RandomForestClassifier()
model.fit(X_train_transformed,Y_train)
```

Prédiction

Entrée []:

```
y_pred = model.predict((X_test_transformed)
data= y_pred
```

RESUME

Les systèmes de service de la vie réelle sont souvent très complexes. Il y a beaucoup de sources d'incertitudes qui rendent leur analyse complexe et difficile.