# Elaboration step

1. **Detailed architecture of the Project by describing all the functionalities and the employed languages;**

**Functionalities :**

Predict the severity of the accident (under the premise of an accident)

Predict the probability of an accident occurring under specified conditions. (Conditions are specified)

Query the accident occurrence map according to the conditions

Various charts

A certain area, time (year, month and day) and the number of accidents

Type of intersection where the accident occurred

Atmospheric conditions and collision types

Road category and transportation system

Slope and surface condition

Near what facilities

Severity of injury

Gender and reasons for travel

Vehicle Type

**The employed languages:**

**Python,HTML,CSS,JS,Nginx**

1. **Scraping and collect the data**

There are 4 CSV files in our dataset, which are caracteristics.csv, places.csv, users.csv and vehicles.csv.

caracteristics.csv:



places.csv:



users.csv:



vehicles.csv:



A detailed description of each column is on the website:

<https://www.kaggle.com/ahmedlahlou/accidents-in-france-from-2005-to-2016>

What we need to do is to predict the probability of an accident based on various environmental factors on the day of the location and to predict the level of casualties based on the accident that has occurred.

So the most useful of the four files for forecasting are caracteristics.csv and places.csv.

First, merge the two files(caracteristics.csv and places.csv) through Num\_Acc.

Second, delete columns that are not related to predicting by pandas.

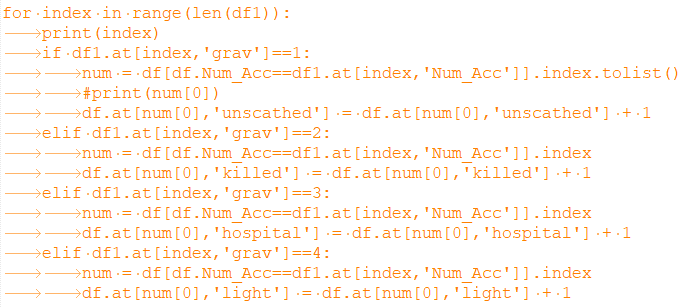
There are many columns in our dataset that contain all aspects of the incident. But some columns don't help relative to the predictions, such as the address and the commune number.And there are several columns in the dataset that don’t have an introduction to it on the site, such as ‘agg’.

So I deleted these columns using pandas and got a new CSV file(by deal.py):



Third, count the number of people with different degrees of injury (unscathed, killed, hospital, light) in the 'users.csv' to the accident (caracteristics.csv').

One attribute in the users.csv file is grav, which marks the injury status of everyone in each accident. An accident usually corresponds to several related people. I need to count the number of people injured in each accident. So I use deal. Py related code to count the number of injured in each accident.



Then the new caracteristics.csv:

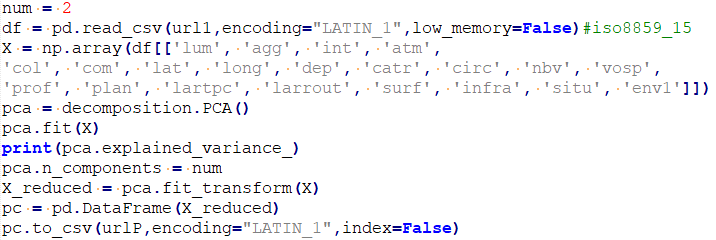


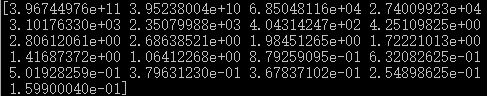
1. **Data cleaning and transformation**

First, delete items with invalid data.

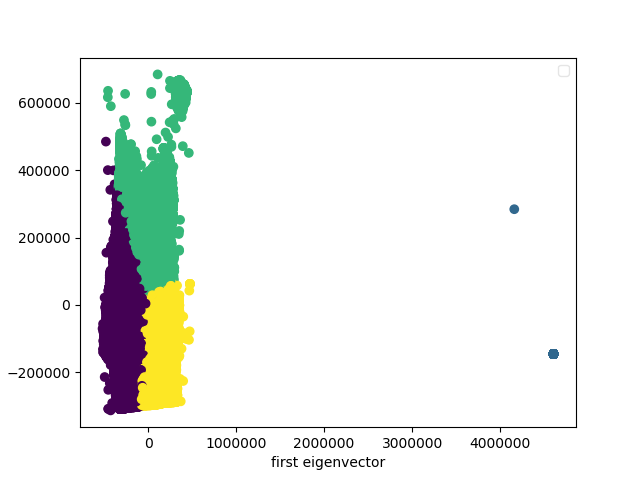
The latitude and longitude column contains a large number of 0 and null, and the other columns also contain a lot of null data. So I need to delete these data by pandas.

Second, Principal component analysis(by PCA.py)





Then we can delete the lowest importance features and make a scatter plot of the data using the first and second eigenvenctor.



Explain what the color means and make a short analysis of the image.

1. **Analysis of the dataset**

Calculate the option that has the most occurrences of each column in the 'caracteristics.csv' by 'calculate.py':

an

7 8815

8 10753

10 22864

11 22323

12 20162

13 20041

14 22163

15 29016

16 40652

Max: 16 40652

mois

1 15947

2 13793

3 15887

4 16586

5 17803

6 18736

7 18099

8 15703

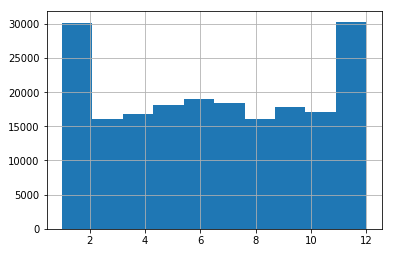
9 17587

10 16900

11 15226

12 14522

Max: 6 18736



lum

1 136709

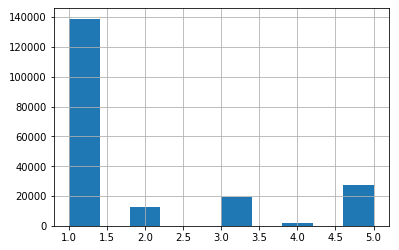
2 12370

3 19239

4 1574

5 26897

Max: 1 136709



int

0 5

1 142774

2 21565

3 16971

4 2636

5 1228

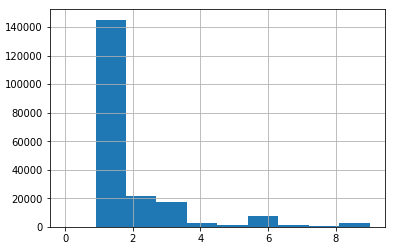
6 7419

7 1324

8 232

9 2635

Max: 1 142774



atm

0 3

1 158527

2 19544

3 4644

4 1152

5 1751

6 561

7 2672

8 6607

9 1328

Max: 1 158527

col

1 24266

2 20939

3 53766

4 5366

5 6718

6 63840

7 21894

Max: 6 63840

catr

1 13421

2 11078

3 86281

4 80901

5 330

6 1508

9 3270

Max: 3 86281

circ

0 11675

1 24498

2 136278

3 23354

4 984

Max: 2 136278

vosp

0 186031

1 3613

2 2775

3 4370

Max: 0 186031

prof

0 17117

1 143090

2 29636

3 4009

4 2937

Max: 1 143090

plan

0 12907

1 143822

2 19579

3 17445

4 3036

Max: 1 143822

lartpc

0 169232

1 61

2 70

3 68

4 68

5 498

6 185

7 142

8 263

9 89

10 3177

11 140

12 263

13 118

14 108

15 5902

...

800 5

811 1

814 1

840 1

900 1

907 1

915 1

Max: 0 169232

larrout

0 67266

1 18

2 2

3 8

4 27

5 43

6 118

7 77

8 27

9 13

10 162

11 9

12 17

13 12

14 12

15 70

16 3

17 11

18 11

19 5

20 247

...

950 1

960 2

970 1

990 3

999 2

Max: 0 67266

surf

0 7362

1 151605

2 33814

3 225

4 68

5 589

6 148

7 1342

8 453

9 1183

Max: 1 151605

infra

0 173874

1 1369

2 2679

3 2608

4 650

5 14146

6 1336

7 127

Max: 0 173874

situ

0 11447

1 163444

2 1105

3 16169

4 3612

5 1012

Max: 1 163444

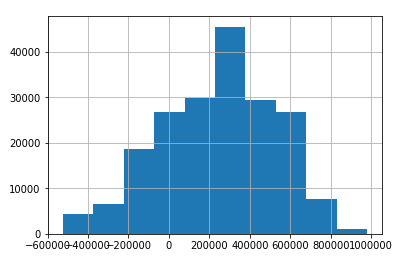
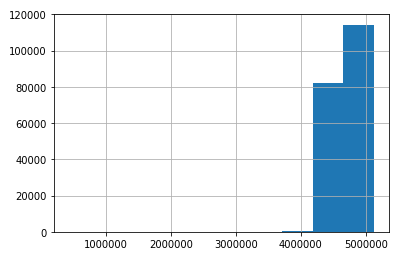
env1

0 101521

3 9705

99 85563

Max: 0 101521



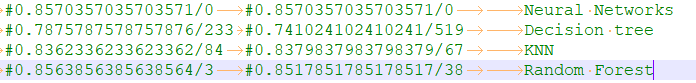
Lat Long

The plotted histograms and measures should be also analysed

In addition to latitude and longitude, the attributes in our data set are discrete rather than continuous.

We use 'start.py' to build different machine learning models to predict the number of different degrees of injuries caused by accidents that have occurred.

Here is graph of the accuracy of predicting the number of deaths and the number of predicted deaths using four models.

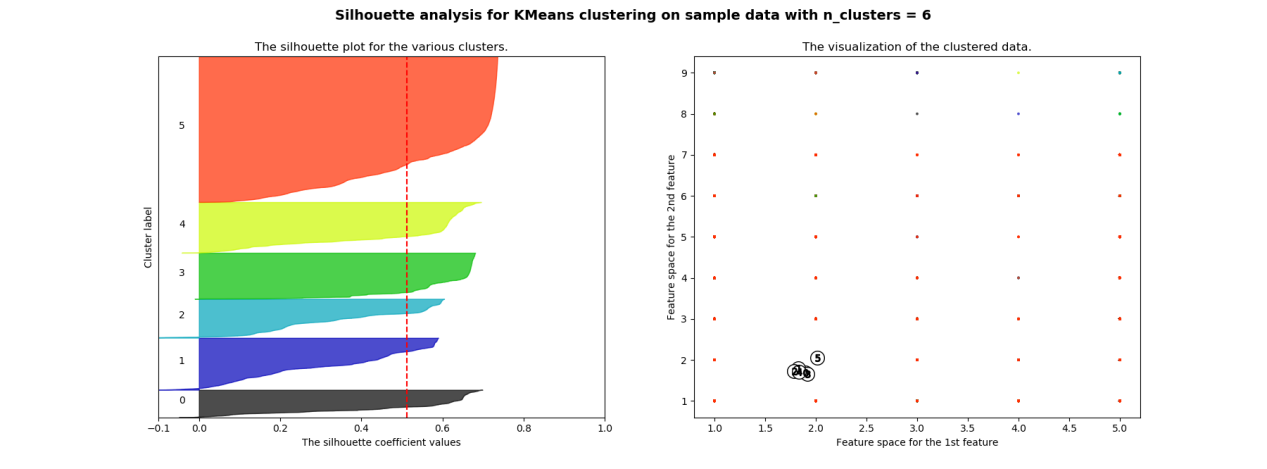
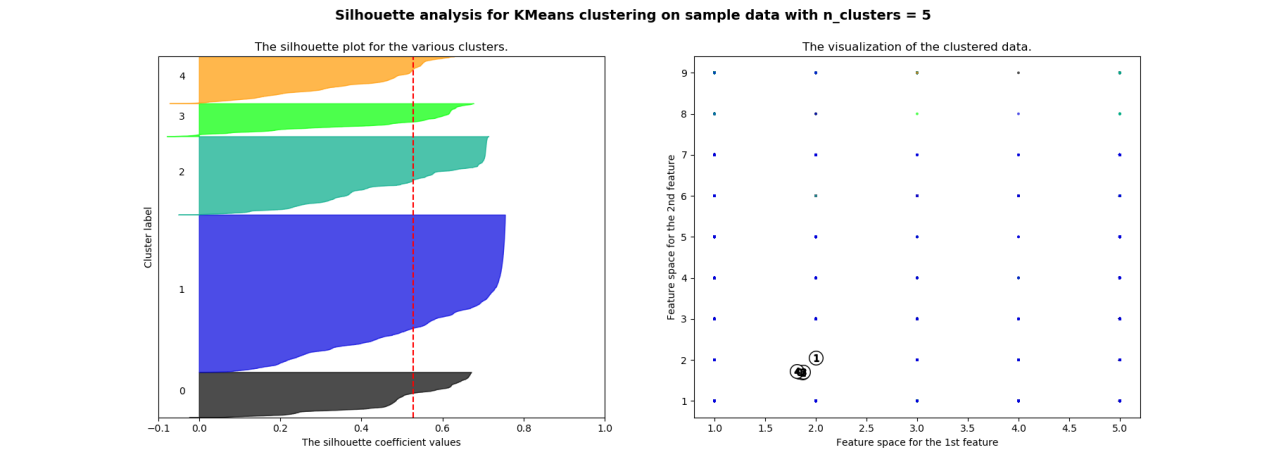
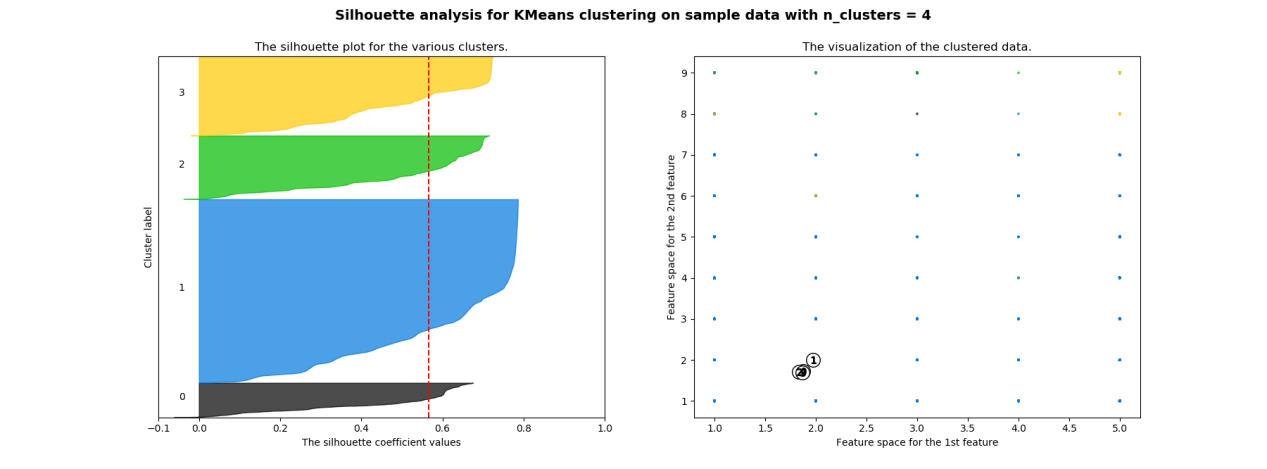
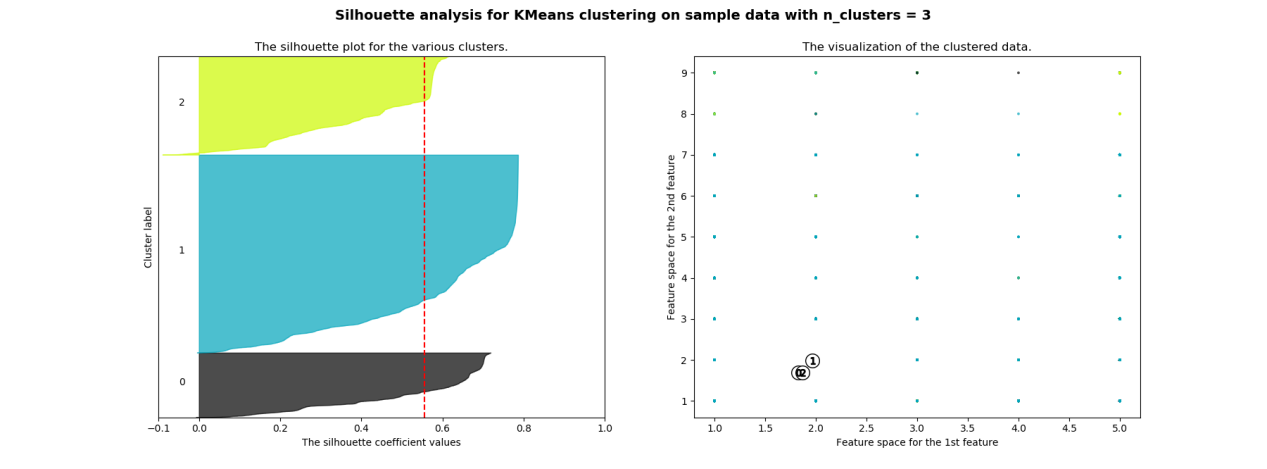
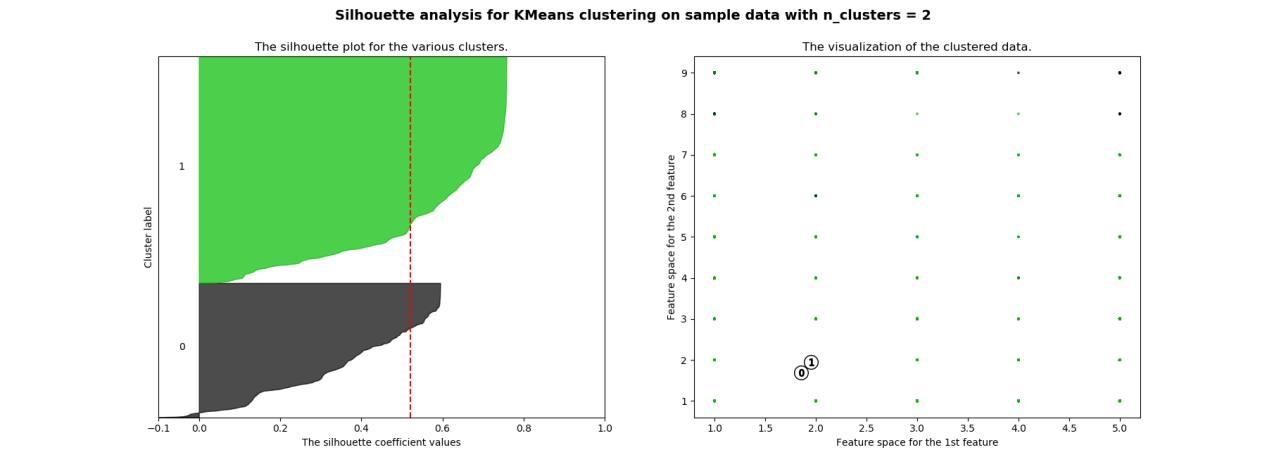


Add some explanations for clustering and Silhuette index.

You can analyze the data in each cluster and try to explain the clustering result.

Clusting(by clusting.py)

Silhuette Analysis(by SilhuetteAnalysis.py)



I don't think these clusters have a particularly good effect. But I think when n\_clusters=3 is best now.