

Assignment2-Advanced Programming

Wanwen LIU (100376272)& Sunhe

In [1]:

```
import pandas as pd
cars = pd.read_csv('auto-mpg.data', delim_whitespace=True, header=None)
cars.describe()
# The response variable is mpg (miles per gallon)
# Attribute 6 is the car model year: 70, 71, 72, ..., 82
modelyear = cars.iloc[:,6].tolist()
mpg = cars.iloc[:,0].tolist()
mymodelyear=list(set(modelyear))
dic1=cars.set_index([6]).T.to_dict('list')

def categorylist(list1,char,list2):
    chPos=[]
    chlist2=[]
    a=0
    try:
        chPos=list((pos for pos,val in enumerate(list1) if(val == char)))
    except:
        pass
    for po in chPos:
        a=list2[po]
        chlist2.append(a)
    return(chlist2)

for i in mymodelyear:
    dic1[i]=str(categorylist(modelyear, i, mpg))
dic1
```

#Second setp:

define a function "categorylist",using the list of medi oyear,element of medi oyear and mpg,find the index of the same element in model year, use the index to find the one-to-one element in mpg,the output of this function is the values of dictionary.
(out[1])

C:\Anaconda3\lib\site-packages\ipykernel_launcher.py:9: UserWarning: DataFrame columns are not unique, some columns will be omitted.

```
if __name__ == '__main__':
```

Out[1]:

```
{70: '[18.0, 15.0, 18.0, 16.0, 17.0, 15.0, 14.0, 14.0, 14.0, 15.0, 15.0, 14.0, 15.0, 14.0, 24.0, 22.0, 18.0, 21.0, 27.0, 26.0, 25.0, 24.0, 25.0, 26.0, 21.0, 10.0, 10.0, 11.0, 9.0]',
 71: '[27.0, 28.0, 25.0, 25.0, 19.0, 16.0, 17.0, 19.0, 18.0, 14.0, 14.0, 14.0, 14.0, 12.0, 13.0, 13.0, 18.0, 22.0, 19.0, 18.0, 23.0, 28.0, 30.0, 30.0, 31.0, 35.0, 27.0, 26.0]',
 72: '[24.0, 25.0, 23.0, 20.0, 21.0, 13.0, 14.0, 15.0, 14.0, 17.0, 11.0, 13.0, 12.0, 13.0, 19.0, 15.0, 13.0, 13.0, 14.0, 18.0, 22.0, 21.0, 26.0, 22.0, 28.0, 23.0, 28.0, 27.0]',
 73: '[13.0, 14.0, 13.0, 14.0, 15.0, 12.0, 13.0, 13.0, 14.0, 13.0, 12.0, 13.0, 18.0, 16.0, 18.0, 18.0, 23.0, 26.0, 11.0, 12.0, 13.0, 12.0, 18.0, 20.0, 21.0, 22.0, 18.0, 19.0, 21.0, 26.0, 15.0, 16.0, 29.0, 24.0, 20.0, 19.0, 15.0, 24.0, 20.0, 11.0]',
 74: '[20.0, 21.0, 19.0, 15.0, 31.0, 26.0, 32.0, 25.0, 16.0, 16.0, 18.0, 16.0, 13.0, 14.0, 14.0, 14.0, 29.0, 26.0, 26.0, 31.0, 32.0, 28.0, 24.0, 26.0, 24.0, 26.0, 31.0]',
 75: '[19.0, 18.0, 15.0, 15.0, 16.0, 15.0, 16.0, 14.0, 17.0, 16.0, 15.0, 18.0, 21.0, 20.0, 13.0, 29.0, 23.0, 20.0, 23.0, 24.0, 25.0, 24.0, 18.0, 29.0, 19.0, 23.0, 23.0, 22.0, 25.0, 33.0]',
 76: '[28.0, 25.0, 25.0, 26.0, 27.0, 17.5, 16.0, 15.5, 14.5, 22.0, 22.0, 24.0, 22.5, 29.0, 24.5, 29.0, 33.0, 20.0, 18.0, 18.5, 17.5, 29.5, 32.0, 28.0, 26.5, 20.0, 13.0, 19.0, 19.0, 16.5, 16.5, 13.0, 13.0, 13.0]',
 77: '[31.5, 30.0, 36.0, 25.5, 33.5, 17.5, 17.0, 15.5, 15.0, 17.5, 20.5, 19.0, 18.5, 16.0, 15.5, 15.5, 16.0, 29.0, 24.5, 26.0, 25.5, 30.5, 33.5, 30.0, 30.5, 22.0, 21.5, 21.5]',
 78: '[43.1, 36.1, 32.8, 39.4, 36.1, 19.9, 19.4, 20.2, 19.2, 20.5, 20.2, 25.1, 20.5, 19.4, 20.6, 20.8, 18.6, 18.1, 19.2, 17.7, 18.1, 17.5, 30.0, 27.5, 27.2, 30.9, 21.1, 23.2, 23.8, 23.9, 20.3, 17.0, 21.6, 16.2, 31.5, 29.5]',
 79: '[21.5, 19.8, 22.3, 20.2, 20.6, 17.0, 17.6, 16.5, 18.2, 16.9, 15.5, 19.2, 18.5, 31.9, 34.1, 35.7, 27.4, 25.4, 23.0, 27.2, 23.9, 34.2, 34.5, 31.8, 37.3, 28.4, 28.8, 26.8, 33.5]',
 80: '[41.5, 38.1, 32.1, 37.2, 28.0, 26.4, 24.3, 19.1, 34.3, 29.8, 31.3, 37.0, 32.2, 46.6, 27.9, 40.8, 44.3, 43.4, 36.4, 30.0, 44.6, 40.9, 33.8, 29.8, 32.7, 23.7, 35.0, 23.6, 32.4]',
 81: '[27.2, 26.6, 25.8, 23.5, 30.0, 39.1, 39.0, 35.1, 32.3, 37.0, 37.7, 34.1, 34.7, 34.4, 29.9, 33.0, 34.5, 33.7, 32.4, 32.9, 31.6, 28.1, 30.7, 25.4, 24.2, 22.4, 26.6, 20.2, 17.6]',
 82: '[28.0, 27.0, 34.0, 31.0, 29.0, 27.0, 24.0, 23.0, 36.0, 37.0, 31.0, 38.0, 36.0, 36.0, 36.0, 34.0, 38.0, 32.0, 38.0, 25.0, 38.0, 26.0, 22.0, 32.0, 36.0, 27.0, 27.0, 44.0, 32.0, 28.0, 31.0]}'
```

In [2]:

```
lenlist=[]
for i in mymodelyear:
    a=len(categorylist(modelyear, i, mpg))
    lenlist.append(a)
lenlist
sumlist=[]
for i in mymodelyear:
    a=sum(categorylist(modelyear, i, mpg))
    sumlist.append(a)
sumlist
index=0
average=[]
index=0
while index<len(lenlist):
    a=sumlist[index]/lenlist[index]
    average.append(a)
    index=index+1
average
```

Out[2]:

```
[17.689655172413794,
 21.25,
 18.714285714285715,
 17.1,
 22.703703703703702,
 20.266666666666666,
 21.573529411764707,
 23.375,
 24.061111111111114,
 25.09310344827585,
 33.696551724137926,
 30.33448275862069,
 31.70967741935484]
```

#the output of the first function has the response variable value of mpg(for each categorical value)
use reduce function to calculate the sum of response variable values and the length for each categorical value.
then calculate the mean value(out[2])

In [3]:

```
import copy
y=copy.deepcopy(modelyear)

for i in mymodelyear:
    dic1[i]=average[i-70]
dic1

def replacelist(list1, char, list2, dic1):
    chPos=[]
    chlist2=[]
    a=0
    try:
        chPos=list((pos for pos, val in enumerate(list1) if (val == char)))
    except:
        pass
    for po in chPos:
        list1[po]=dic1[char]
    return(list1)

for i in mymodelyear:
    replacelist(y, i, average, dic1)

y
```

#step4, create a copy of the response variable values, replace the output of Step 3 (the average of response variable values for each categorical value), to renew the dictionary, Define the second function "replacelist", replace the response variable values with the average value for each categorical value.

[illegible]

<http://localhost:8888/nbconvert/html/Untitled7.ipynb?download=false>

17. 1,

17. 1,

17. 1,

17. 1,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

22. 703703703703702,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

20. 2666666666666666,

[illegible]

[illegible]

<http://localhost:8888/nbconvert/html/Untitled7.ipynb?download=false>

In []: