Lab 08 - DataFrame

DataFrame is a data structure provided by pandas library. It is a 2- dimensional structure & can be compared to a table of rows and columns.

Each row can be identified by an integer index (0..N) or a label explicitly set when creating a DataFrame object. Each column can be of distinct type and is identified by a label.

Create a sample DataFrame

Create a DataFrame from a dictionary, containing two columns: numbers and colors. Each key represent a column name and the value is a series of data, the content of the column:

```
import pandas as pd
df = pd.DataFrame({'numbers': [1, 2, 3], 'colors': ['red', 'white', 'blue']})
```

Show contents of dataframe:

```
print(df)
```

```
   numbers colors
0        1    red
1        2  white
2        3   blue
```

To specify the order, use the columns parameter.

```
df = pd.DataFrame({'numbers': [1, 2, 3], 'colors': ['red', 'white', 'blue']},
                  columns=['numbers', 'colors'])
print(df)
```

```
   numbers colors
0        1    red
1        2  white
2        3   blue
```

Create a sample DataFrame using Numpy

```
import numpy as np
import pandas as pd
```

```
df = pd.DataFrame(np.random.randn(5, 3), columns=['N1', 'N2', 'N3'])
print(df)
```

```
          N1        N2        N3
0  0.154947  0.378163 -0.887786
1 -1.980796 -0.347912  0.156349
2  1.230291  1.202380 -0.387327
3 -0.302303 -1.048553 -1.420018
```

Create a DataFrame from a dictionary of lists

```
import pandas as pd
df = pd.DataFrame({'N1' : [1, 2, 3, 4], 'N2' : [4, 3, 2, 1]})
print(df)
```

```
   N1  N2
0   1   4
1   2   3
2   3   2
3   4   1
```

Create a DataFrame from a list of dictionaries

```
import pandas as pd
L = [{'Name': 'John', 'Last Name': 'Smith'},
     {'Name': 'Mary', 'Last Name': 'Wood'}]
df = pd.DataFrame(L)
print(df)
```

```
   Name Last Name
0  John     Smith
1  Mary      Wood
```

Input from Text Files

Create a file sample.txt with the following content

```
1, 2.1, 0.1
2, 4.5, 9.0
3, 5.5, 2.1
4, 2.3, 3.3
5, 7.7, 8.8
```

```
# Load data from file
data = np.loadtxt("sample.txt", delimiter=',')
print(data)
```

```
[[1.  2.1 0.1]
 [2.  4.5 9. ]
 [3.  5.5 2.1]
 [4.  2.3 3.3]
 [5.  7.7 8.8]]
```

An alternative way to read file as a cvs file.

```
data = pd.read_csv("sample.txt", delimiter=',')
print(data)
```

```
   1   2.1   0.1
0  2   4.5   9.0
1  3   5.5   2.1
2  4   2.3   3.3
3  5   7.7   8.8
```

Complex text-files with header. Create a file sample.txt with the following content.

```
ID,Weight,Score
1,2.1,0.1
2,4.5,9.0
3,5.5,2.1
4,2.3,3.3
5,7.7,8.8
```

```
data = pd.read_csv('sample1.txt', delimiter=',')
```

```
print(data)
print("Print column Score")
print(data.Score)
```

```
   ID  Weight  Score
0   1     2.1    0.1
1   2     4.5    9.0
2   3     5.5    2.1
3   4     2.3    3.3
4   5     7.7    8.8
Print column Score
0    0.1
1    9.0
2    2.1
3    3.3
4    8.8
Name: Score, dtype: float64
```

Access data

Sometimes Pandas DataFrames are very large that it is impossible to see all the data at the same time. So for this, there are several methods to access the data.

```
df = pd.DataFrame(np.random.randn(10, 3), columns=['N1', 'N2', 'N3'])
print(df)
```

```
          N1        N2        N3
0 -0.386871 -0.510293  0.183925
1 -0.385490 -1.601836 -0.887181
2 -0.932789  1.243319  0.812674
3  0.587259 -0.505358 -0.815792
4 -0.507518 -1.051880  2.497200
5 -2.245322  0.564009 -1.284552
6 -0.104343 -0.988002 -1.177629
7 -1.140196  1.754986 -0.132988
8 -0.765702  0.555787  0.010349
9  0.720034 -1.824257  0.303604
```

The head(n) method is used to show first n rows (n is optional its default value is 5)

```
df.head(5)
```

	N1	N2	N3
0	-0.386871	-0.510293	0.183925
1	-0.385490	-1.601836	-0.887181
2	-0.932789	1.243319	0.812674
3	0.587259	-0.505358	-0.815792
4	-0.507518	-1.051880	2.497200

If you want to see the last n rows you can use the tail() function

```
df.tail(5)
```

	N1	N2	N3
5	-2.245322	0.564009	-1.284552
6	-0.104343	-0.988002	-1.177629
7	-1.140196	1.754986	-0.132988
8	-0.765702	0.555787	0.010349
9	0.720034	-1.824257	0.303604

For accessing the specific columns you can specify the column name in “[]” brackets.

```
N2 = df['N2']
print(N2)
```

```
0   -0.510293
1   -1.601836
2    1.243319
3   -0.505358
4   -1.051880
5    0.564009
6   -0.988002
7    1.754986
8    0.555787
9   -1.824257
Name: N2, dtype: float64
```

To select multiple columns, you need to pass the array/list of the feature names.

```
df[['N1','N2']]
```

	N1	N2
0	-0.386871	-0.510293
1	-0.385490	-1.601836
2	-0.932789	1.243319
3	0.587259	-0.505358
4	-0.507518	-1.051880
5	-2.245322	0.564009
6	-0.104343	-0.988002
7	-1.140196	1.754986
8	-0.765702	0.555787
9	0.720034	-1.824257

For accessing the number of rows we can get it just like we get it from the python list.

```
print(df[1:-2])
```

```
          N1        N2        N3
1 -0.385490 -1.601836 -0.887181
2 -0.932789  1.243319  0.812674
3  0.587259 -0.505358 -0.815792
4 -0.507518 -1.051880  2.497200
5 -2.245322  0.564009 -1.284552
6 -0.104343 -0.988002 -1.177629
7 -1.140196  1.754986 -0.132988
```

Exercise

1. Exploratory Data Analysis for IRIS dataset

- Read dataset iris from file iris.csv
- Show the first 5 data points.
- Compute number of data (count), mean, standard deviation (std), min and max. Create a dataframe to show your result as below. (DO NOT USE Build-in functions to compute)

```
      sepal_length    sepal_width    petal_length    petal_width
count         ?                ?              ?               ?
mean          ?                ?              ?               ?
std           ?                ?              ?               ?
min           ?                ?              ?               ?
max           ?                ?              ?               ?
```

2. Exploratory Data Analysis for POPULATION dataset

- Read dataset population from file population.csv
- Show the first 5 data points.
- Compute number of data (count), mean, standard deviation (std), min and max by year. Create a dataframe to show your result as below. (DO NOT USE Build-in functions to compute

```
      Country Name    Country Code    Mean Std Min Max
0     Arab World      ARB              ?    ?   ?   ?
1     Portugal        PRT           ...
.......
```