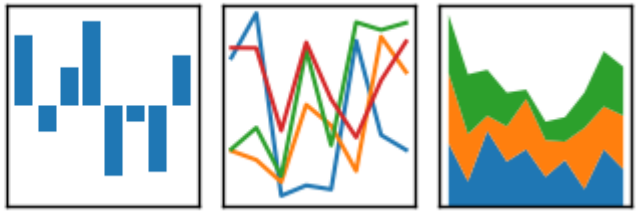
pandas

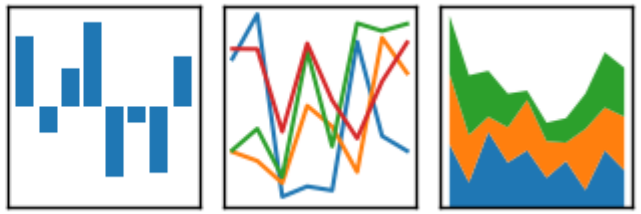
$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$



- "Pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the Python programming language." - https://pandas.pydata.org/

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$



- Some data can be represented by Pandas DataFrame and Series (tables).
- Load data from a file
- Modify a DataFrame and save it to a file
- Combine two DataFrames into one DataFrame
- Handle Missing Data

import pandas as pd

A visualization of **Dataframe**

```
data = pd.read_csv('buy_gpu.csv')
data
```

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160

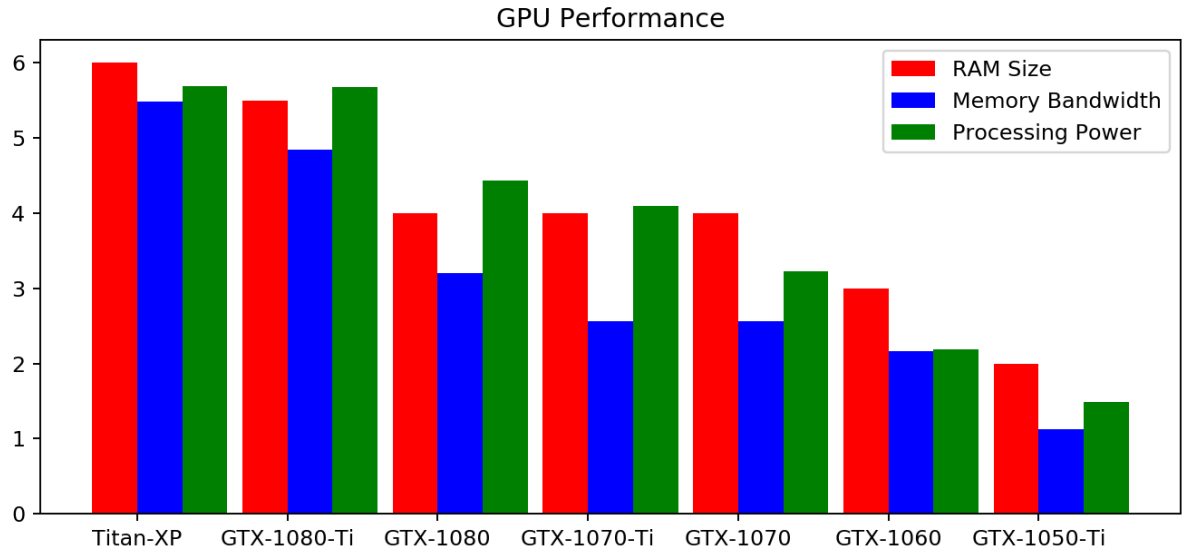
```
type(data)
```

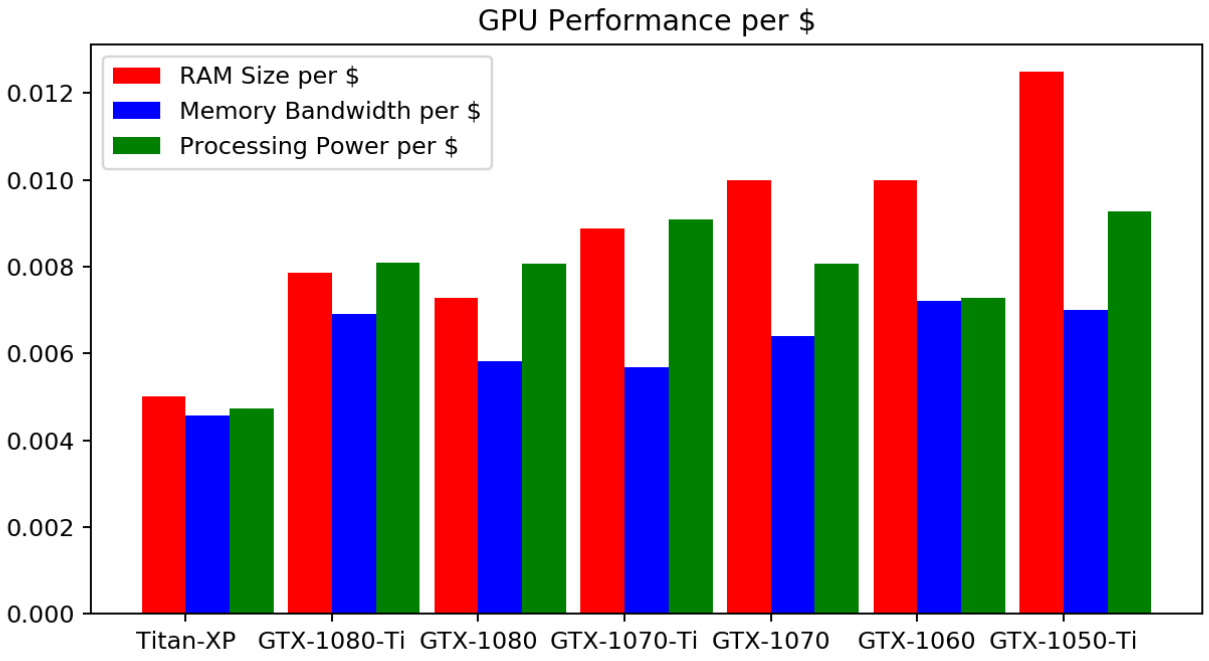
```
pandas.core.frame.DataFrame
```

- Visualize the data using bar plot

DataFrame in Pandas

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160





index of the first column

index of the last column

index of the first row

index of the second row

index of the last row

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160

an index can be an integer or a string or other object

https://pandas.pydata.org/pandas-docs/stable/generated/pandas.Index.html

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160

Get a column by its name/index

```
GPU_Name=data['GPU_Name']
GPU_Name
```

```
0       Titan-XP
1    GTX-1080-Ti
2       GTX-1080
3    GTX-1070-Ti
4       GTX-1070
5       GTX-1060
6    GTX-1050-Ti
Name: GPU_Name, dtype: object
```

```
type(GPU_Name)
```

```
pandas.core.series.Series
```

get an element of the series by index

```
GPU_Name[1]
```

```
'GTX-1080-Ti'
```

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160

get an element by index

```
row1[4]
```

```
700
```

get a row by the integer index

```
row1 = data.iloc[1,:]
row1
```

```
GPU_Name              GTX-1080-Ti
RAM_Size                       11
Memory_Bandwidth              484
Processing_Power          5.66989
Price                         700
Name: 1, dtype: object
```

```
type(row1)
```

```
pandas.core.series.Series
```

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160

get an element by index

```
price[0]
```

```
1200
```

get a column by the integer index

```
price = data.iloc[:,4]
price
```

```
0    1200
1     700
2     550
3     450
4     400
5     300
6     160
Name: Price, dtype: int64
```

```
type(price)
```

```
pandas.core.series.Series
```

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160

```
data_np = data.values # convert a dataframe to a numpy array (2D)
type(data)
```

```
pandas.core.frame.DataFrame
```

```
data_np # a 2D array is a sequence of 1D arrays
```

```
array([['Titan-XP', 12, 547.7, 5.6832, 1200],
       ['GTX-1080-Ti', 11, 484.0, 5.669887999999999, 700],
       ['GTX-1080', 8, 320.0, 4.43648, 550],
       ['GTX-1070-Ti', 8, 256.0, 4.093056, 450],
       ['GTX-1070', 8, 256.0, 3.23136, 400],
       ['GTX-1060', 6, 216.0, 2.18624, 300],
       ['GTX-1050-Ti', 4, 112.0, 1.483776, 160]], dtype=object)
```

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160

```
data_np = data.iloc[:,1:5].values # convert a dataframe to a numpy array (2D)
data_np
```

```
array([[  12.      ,  547.7     ,    5.6832  , 1200.      ],
       [  11.      ,  484.     ,    5.669888,  700.      ],
       [   8.      ,  320.     ,    4.43648 ,  550.      ],
       [   8.      ,  256.     ,    4.093056,  450.      ],
       [   8.      ,  256.     ,    3.23136 ,  400.      ],
       [   6.      ,  216.     ,    2.18624 ,  300.      ],
       [   4.      ,  112.     ,    1.483776,  160.      ]])
```

```
data_np.dtype
```

```
dtype('float64')
```

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1200
1	GTX-1080-Ti	11	484.0	5.669888	700
2	GTX-1080	8	320.0	4.436480	550
3	GTX-1070-Ti	8	256.0	4.093056	450
4	GTX-1070	8	256.0	3.231360	400
5	GTX-1060	6	216.0	2.186240	300
6	GTX-1050-Ti	4	112.0	1.483776	160

```
price = data.iloc[:,4].values   # convert a series to a numpy array (1D)
type(price)
```

```
numpy.ndarray
```

```
price
```

```
array([1200,  700,  550,  450,  400,  300,  160], dtype=int64)
```

modify the Dataframe and save it to a csv file

```
data['Price'] *= 1.2
```

```
data
```

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1440.0
1	GTX-1080-Ti	11	484.0	5.669888	840.0
2	GTX-1080	8	320.0	4.436480	660.0
3	GTX-1070-Ti	8	256.0	4.093056	540.0
4	GTX-1070	8	256.0	3.231360	480.0
5	GTX-1060	6	216.0	2.186240	360.0
6	GTX-1050-Ti	4	112.0	1.483776	192.0

```
data.to_csv('gpu_info_new.csv', index=False, sep=',')
```

axis of a Dataframe

axis 1

	GPU_Name	RAM_Size	Memory_Bandwidth	Processing_Power	Price
0	Titan-XP	12	547.7	5.683200	1440.0
1	GTX-1080-Ti	11	484.0	5.669888	840.0
2	GTX-1080	8	320.0	4.436480	660.0
3	GTX-1070-Ti	8	256.0	4.093056	540.0
4	GTX-1070	8	256.0	3.231360	480.0
5	GTX-1060	6	216.0	2.186240	360.0
6	GTX-1050-Ti	4	112.0	1.483776	192.0

axis 0

data['GPU_Name'] is the column-0

data.iloc[i, j] is an element of the table

data.iloc[i], data.iloc[i,:] and data.loc[i] refer to the row-i

Show Me More about Pandas

Pandas_basics.ipynb

Pandas_advanced.ipynb

Pandas_data_editing_example.ipynb

Pandas_missing_value_example.ipynb