IPA 주관 인공지능센터 기본(fundamental) 과정

- GitHub link: here
- E-Mail: windkyle7@gmail.com

이번 장에서는 `titanic` 데이터로 데이터 분석을 진행하고자 한다.

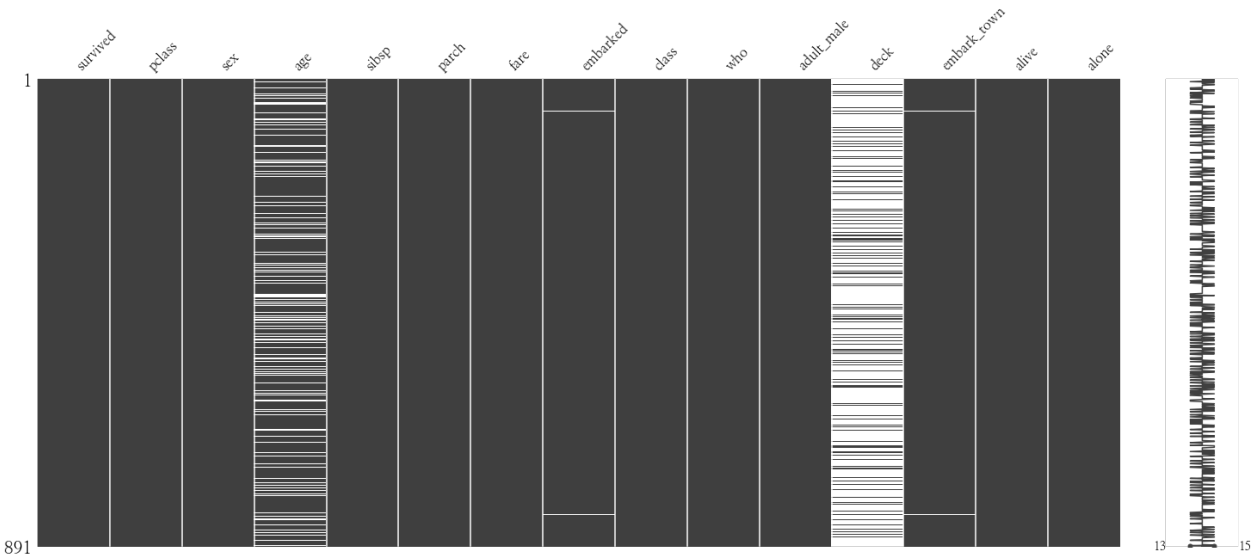
In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import missingno as mino
import matplotlib.font_manager as fm
fm.rcParams['font.family'] = 'NanumMyeongjo'
```

In [2]:
```
titanic = sns.load_dataset('titanic')
```

불러온 데이터에 missing value가 있는 것을 확인할 수 있다.

In [3]:
```
%matplotlib inline
mino.matrix(titanic)
```

Out[3]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8a39944048>
```



`titanic` 데이터에 대한 정보를 살펴보면 다음과 같다.

In [4]:
```
import pandas_profiling as pp
pp.ProfileReport(titanic)
```

Out[4]:

Overview

Dataset info

Number of variables	15
Number of observations	891
Total Missing (%)	6.5%
Total size in memory	80.6 KiB
Average record size in memory	92.6 B

Variables types

Numeric	5
Categorical	7
Boolean	3
Date	0
Text (Unique)	0
Rejected	0
Unsupported	0

Warnings

- `age` has 177 / 19.9% missing values Missing
- `deck` has 688 / 77.2% missing values Missing
- `fare` has 15 / 1.7% zeros Zeros
- `parch` has 678 / 76.1% zeros Zeros
- `sibsp` has 608 / 68.2% zeros Zeros
- Dataset has 107 duplicate rows Warning

Variables

adult_male
Boolean

Distinct count	2
Unique (%)	0.2%
Missing (%)	0.0%
Missing (n)	0

Mean	0.60269

True	537
False	354

Toggle details

age
Numeric

Distinct count	89
Unique (%)	10.0%
Missing (%)	19.9%
Missing (n)	177
Infinite (%)	0.0%
Infinite (n)	0

Mean	29.699
Minimum	0.42
Maximum	80
Zeros (%)	0.0%

Toggle details

alive
Categorical

Distinct count	2
Unique (%)	0.2%
Missing (%)	0.0%
Missing (n)	0

no	549
yes	342

Toggle details

alone
Boolean

Distinct count	2
Unique (%)	0.2%
Missing (%)	0.0%
Missing (n)	0

Mean	0.60269

True	537
False	354

Toggle details

class
Categorical

Distinct count	3
Unique (%)	0.3%
Missing (%)	0.0%
Missing (n)	0

Third	491
First	216
Second	184

Toggle details

deck
Categorical

Distinct count	8
Unique (%)	0.9%
Missing (%)	77.2%
Missing (n)	688

C	59
B	47
D	33
Other values (4)	64
(Missing)	688

Toggle details

embark_town
Categorical

Distinct count	4
Unique (%)	0.4%
Missing (%)	0.2%
Missing (n)	2

Southampton	644
Cherbourg	168
Queenstown	77
(Missing)	2

Toggle details

embarked
Categorical

Distinct count	4
Unique (%)	0.4%
Missing (%)	0.2%
Missing (n)	2

S	644
C	168
Q	77
(Missing)	2

Toggle details

fare
Numeric

Distinct count	248
Unique (%)	27.8%
Missing (%)	0.0%
Missing (n)	0
Infinite (%)	0.0%
Infinite (n)	0

Mean	32.204
Minimum	0
Maximum	512.33
Zeros (%)	1.7%

Toggle details

parch
Numeric

Distinct count	7
Unique (%)	0.8%
Missing (%)	0.0%
Missing (n)	0
Infinite (%)	0.0%
Infinite (n)	0

Mean	0.38159
Minimum	0
Maximum	6
Zeros (%)	76.1%

Toggle details

pclass
Numeric

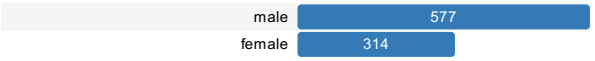
Distinct count	3
Unique (%)	0.3%
Missing (%)	0.0%
Missing (n)	0
Infinite (%)	0.0%
Infinite (n)	0

Mean	2.3086
Minimum	1
Maximum	3
Zeros (%)	0.0%

Toggle details

sex
Categorical

Distinct count	2
Unique (%)	0.2%
Missing (%)	0.0%
Missing (n)	0



Toggle details

sibsp
Numeric

Distinct count	7
Unique (%)	0.8%
Missing (%)	0.0%
Missing (n)	0
Infinite (%)	0.0%
Infinite (n)	0

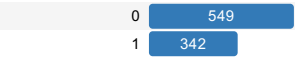
Mean	0.52301
Minimum	0
Maximum	8
Zeros (%)	68.2%

Toggle details

survived
Boolean

Distinct count	2
Unique (%)	0.2%
Missing (%)	0.0%
Missing (n)	0

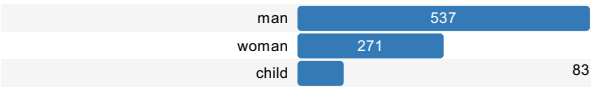
Mean	0.38384



Toggle details

who
Categorical

Distinct count	3
Unique (%)	0.3%
Missing (%)	0.0%
Missing (n)	0



Toggle details

Correlations

Sample

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True

missing value를 `fillna` 를 통해 평균값으로 대체시킨다.

In [5]:
```
titanic.fillna(method='bfill', inplace=True)
titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
survived       891 non-null int64
pclass         891 non-null int64
sex            891 non-null object
age            891 non-null float64
sibsp          891 non-null int64
parch          891 non-null int64
fare           891 non-null float64
embarked       891 non-null object
class          891 non-null category
who            891 non-null object
adult_male     891 non-null bool
deck           890 non-null category
embark_town    891 non-null object
alive          891 non-null object
alone          891 non-null bool
```

```
alone          891 non-null bool
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.6+ KB
```

`sex` 와 `class` 간 생존 여부를 분석해본다.

In [6]:
```
group = titanic.groupby(['sex','class']).survived
total = group.sum()
```

In [7]:
```
total
```

Out[7]:
```
sex     class
female  First     91
        Second    70
        Third     72
male    First     45
        Second    17
        Third     47
Name: survived, dtype: int64
```

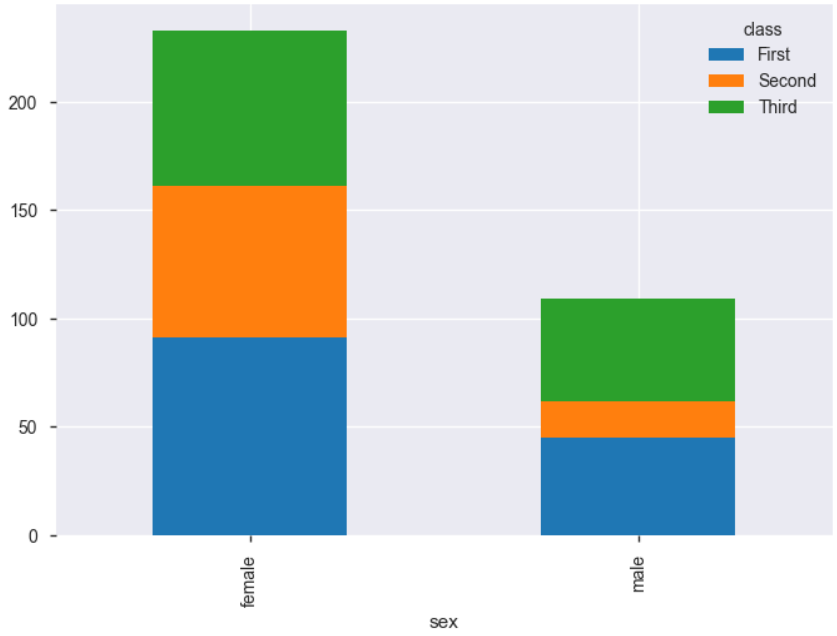
In [8]:
```
total.unstack()
```

Out[8]:

class	First	Second	Third
sex			
female	91	70	72
male	45	17	47

In [9]:
```
total.unstack().plot.bar(stacked=True)
```

Out[9]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8a5fe0ef98>
```



In [10]:
```
titanic.embarked.value_counts()
```

Out[10]:
```
S    645
C    169
Q     77
Name: embarked, dtype: int64
```

In [11]:
```
titanic.embarked.map({'S':0,'C':1,'Q':2})
```

Out[11]:
```
0      0
1      1
2      0
3      0
4      0
5      2
6      0
7      0
8      0
9      1
10     0
11     0
12     0
13     0
14     0
15     0
16     2
17     0
18     0
19     1
20     0
21     0
22     2
23     0
```

```
23    0
24    0
25    0
26    1
27    0
28    2
29    0
     ..
861   0
862   0
863   0
864   0
865   0
866   1
867   0
868   0
869   0
870   0
871   0
872   0
873   0
874   1
875   1
876   0
877   0
878   0
879   1
880   0
881   0
882   0
883   0
884   0
885   2
886   0
887   0
888   0
889   1
890   2
Name: embarked, Length: 891, dtype: int64
```

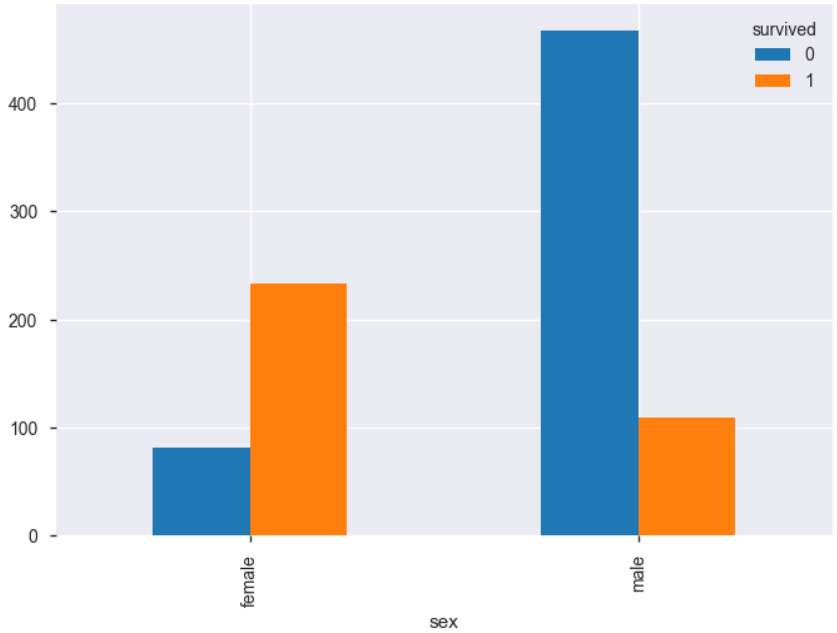
In [12]:
```
titanic.pivot_table('survived','sex',aggfunc=sum)
```

Out[12]:

sex	survived
female	233
male	109

In [13]:
```
table = pd.crosstab(titanic.sex, titanic.survived)
table.plot.bar()
```

Out[13]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f8a5fe0e630>
```



In [14]:
```
table.stack()
```

Out[14]:
```
sex     survived
female  0            81
        1           233
male    0           468
        1           109
dtype: int64
```

In [15]:
```
table.unstack()
```

Out[15]:
```
survived  sex
0         female     81
```

```
          male      468
1         female    233
          male      109
dtype: int64
```

In [16]:
```
survived_group = titanic.groupby(['sex','survived'])
total = survived_group.sum().unstack()
total
```

Out[16]:

	pclass		age		sibsp		parch		fare		adult_male		alone	
survived	0	1	0	1	0	1	0	1	0	1	0	1	0	1
sex														
female	231	447	2112.00	6886.42	98	120	84	120	1864.9752	12101.6876	0.0	0.0	27.0	99.0
male	1159	220	14637.83	2978.42	206	42	97	39	10277.7447	4449.5418	449.0	88.0	347.0	64.0

In [17]:
```
total.stack()
```

Out[17]:

		pclass	age	sibsp	parch	fare	adult_male	alone
sex	survived							
female	0	231	2112.00	98	84	1864.9752	0.0	27.0
	1	447	6886.42	120	120	12101.6876	0.0	99.0
male	0	1159	14637.83	206	97	10277.7447	449.0	347.0
	1	220	2978.42	42	39	4449.5418	88.0	64.0

In [18]:
```
total.plot.bar(stacked=True)
```

Out[18]: <matplotlib.axes._subplots.AxesSubplot at 0x7f8a5fcc9390>