실전 데이터 알고리즘 적용

- 데이터 셋 : https://www.kaggle.com/competitions/titanic/ (https://www.kaggle.com/competitions/titanic/)
- 적용 알고리즘 : PCA

01 라이브러리 및 데이터 불러오기

In [1]:

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
```

In [2]:

```
tr = pd.read_csv("./titanic/train.csv")
test = pd.read_csv("./titanic/test.csv")

tr.shape, test.shape
```

Out[2]:

```
((891, 12), (418, 11))
```

02 train 데이터 셋을 활용한 PCA 알고리즘 적용 및 활용

In [3]:

```
tr.columns
```

Out[3]:

```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

기본 전처리

In [4]:

```
tr.head()
```

Out[4]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	C
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	

In [5]:

```
tr.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

결측치 처리

In [6]:

```
tr['Embarked'].value_counts()
```

Out[6]:

```
S    644
C    168
Q     77
Name: Embarked, dtype: int64
```

In [7]:

```
tr.loc[ tr['Age'].isnull(), 'Age'] =  tr['Age'].mean()
tr.loc[ tr['Embarked'].isnull(), 'Embarked'] ='S'

tr.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   PassengerId  891 non-null    int64
 1   Survived     891 non-null    int64
 2   Pclass       891 non-null    int64
 3   Name         891 non-null    object
 4   Sex          891 non-null    object
 5   Age          891 non-null    float64
 6   SibSp        891 non-null    int64
 7   Parch        891 non-null    int64
 8   Ticket       891 non-null    object
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object
 11  Embarked     891 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

In [8]:

```
di_sex = {"male":0, "female":1}
di_Embarked = {"C":0, "S":1, "Q":2}

tr['Sex_lbl'] = tr['Sex'].map(di_sex)
tr['Embarked_lbl'] = tr['Embarked'].map(di_Embarked)

tr.head()
```

Out[8]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	C
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	

기본적인 전처리가 필요없는 피처를 선택

In [9]:

```
X = tr.drop(['Survived', 'Name', 'Ticket', 'Cabin', 'Sex', 'Embarked'], axis=1)
y = tr['Survived']

print( X.shape, y.shape )
```

```
(891, 8) (891,)
```

In [10]:

```
X.head()
```

Out[10]:

	PassengerId	Pclass	Age	SibSp	Parch	Fare	Sex_lbl	Embarked_lbl
0	1	3	22.0	1	0	7.2500	0	1
1	2	1	38.0	1	0	71.2833	1	0
2	3	3	26.0	0	0	7.9250	1	1
3	4	1	35.0	1	0	53.1000	1	1
4	5	3	35.0	0	0	8.0500	0	1

- 현재 변수가 8개, 이를 PCA를 이용하여 3개의 피처를 갖는 주성분으로 만들어보자.

In [11]:

```
### 데이터 나누기
scaler = StandardScaler()
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                test_size=0.3, random_state=42)

X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

In [14]:

```
from sklearn.decomposition import PCA

pca = PCA(n_components=2)
scaler = StandardScaler()

X_train = pca.fit_transform(X_train)
X_test = pca.transform(X_test)

X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

In [15]:

```
plt.figure(figsize=(8,6))
plt.scatter(X_train[:,0],X_train[:,1],c=y_train,cmap='plasma')
plt.xlabel('First principal component')
plt.ylabel('Second Principal Component')
```

Out[15]:

```
Text(0, 0.5, 'Second Principal Component')
```