불균형 데이터 처리

- 클래스 간 데이터 분포가 크게 다른 데이터셋에서 모델의 성능을 개선하기 위해 사용
- 오버샘플링 방법 : 소수 클래스의 데이터를 반복적으로 추가하여 데이터의 균형을 맞춤
  - RamdomOverSampler, SMOTE 방법 등
  - 과적합의 위험 존재
- 다운샘플링 방법 : 다수 클래스의 데이터를 줄여 데이터의 균형을 맞춤
  - RandomUnderSampler
  - 정보 손실 위험 존재

```
import seaborn as sns
import pandas as pd

titanic = sns.load_dataset('titanic')

titanic
```

	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
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
886	0	2	male	27.0	0	0	13.0000	S	Second	man	True	NaN	Southampton	no	True
887	1	1	female	19.0	0	0	30.0000	S	First	woman	False	B	Southampton	yes	True
888	0	3	female	NaN	1	2	23.4500	S	Third	woman	False	NaN	Southampton	no	False
889	1	1	male	26.0	0	0	30.0000	C	First	man	True	C	Cherbourg	yes	True
890	0	3	male	32.0	0	0	7.7500	Q	Third	man	True	NaN	Queenstown	no	True

891 rows × 15 columns

```
data = titanic[['survived', 'pclass','sex','age','sibsp','parch','fare','embarked']].dropna()


cat_df = data.select_dtypes(include = ['object']).columns
cat_df
data = pd.get_dummies(data, columns=cat_df)


X = data.drop('survived', axis = 1)
y = data['survived']



# 데이터 분할
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size = 0.2, random_state = 42, stratify= y)
pd.Series(y_train).value_counts()
```

```
survived
0    339
1    230
Name: count, dtype: int64
```

```
# 랜덤오버샘플링 적용

from imblearn.over_sampling import RandomOverSampler

ros = RandomOverSampler(random_state=42)
X_res, y_res = ros.fit_resample(X_train, y_train)

pd.Series(y_res).value_counts()
```

```
survived
1    339
```

```
0    339
Name: count, dtype: int64
```

```
# SMOTE
from imblearn.over_sampling import SMOTE

smote = SMOTE(random_state = 42)
X_res_sm, y_res_sm = smote.fit_resample(X_train, y_train)

pd.Series(y_res_sm).value_counts()
```

```
survived
1    339
0    339
Name: count, dtype: int64
```

```
from imblearn.under_sampling import RandomUnderSampler
rus = RandomUnderSampler(random_state = 42)
X_res_ud , y_res_ud = rus.fit_resample(X_train, y_train)

pd.Series(y_res_ud).value_counts()
```

```
survived
0    230
1    230
Name: count, dtype: int64
```

코딩을 시작하거나 AI로 코드를 생성하세요.