Day 15 | KNN Imputer

Import Library

In [1]:

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.impute import KNNImputer,SimpleImputer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

Import Dataset

In [2]:

```
df = pd.read_csv('train.csv')[['Age','Pclass','Fare','Survived']]
```

In [3]:

```
df.head()
```

Out[3]:

	Age	Pclass	Fare	Survived
0	22.0	3	7.2500	0
1	38.0	1	71.2833	1
2	26.0	3	7.9250	1
3	35.0	1	53.1000	1
4	35.0	3	8.0500	0

Check Missing Value

In [4]:

```
df.isnull().mean() * 100
```

Out[4]:

```
Age         19.86532
Pclass       0.00000
Fare         0.00000
Survived     0.00000
dtype: float64
```

Define X & Y

In [5]:

```
X = df.drop(columns=['Survived'])
y = df['Survived']
```

Train Test Split

In [6]:

```
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=2)
```

```
In [7]: X_train
```

```
Out[7]:
       Age  Pclass      Fare
30    40.0       1   27.7208
10     4.0       3   16.7000
873   47.0       3    9.0000
182    9.0       3   31.3875
876   20.0       3    9.8458
...    ...     ...       ...
534   30.0       3    8.6625
584    NaN       3    8.7125
493   71.0       1   49.5042
527    NaN       1  221.7792
168    NaN       1   25.9250

712 rows × 3 columns
```

Apply KNN Imputer

```
In [19]: knn = KNNImputer(n_neighbors=10,weights='distance')
         X_train_trf = knn.fit_transform(X_train)
         X_test_trf = knn.transform(X_test)
```

Convert in Data Frame

```
In [16]: pd.DataFrame(X_train_trf,columns=X_train.columns)
```

```
Out[16]:
           Age  Pclass      Fare
0    40.000000     1.0   27.7208
1     4.000000     3.0   16.7000
2    47.000000     3.0    9.0000
3     9.000000     3.0   31.3875
4    20.000000     3.0    9.8458
...        ...     ...       ...
707  30.000000     3.0    8.6625
708  26.151292     3.0    8.7125
709  71.000000     1.0   49.5042
710  32.666667     1.0  221.7792
711  49.762895     1.0   25.9250

712 rows × 3 columns
```

In [17]:

```
pd.DataFrame(X_train_trf,columns=X_train.columns).sample(10)
```

Out[17]:

	Age	Pclass	Fare
607	63.0	3.0	9.5875
394	52.0	1.0	30.5000
537	16.0	2.0	10.5000
694	22.0	3.0	7.8958
373	19.0	2.0	36.7500
364	19.0	2.0	13.0000
359	21.0	3.0	7.7500
583	16.0	3.0	8.0500
460	33.0	3.0	15.8500
160	22.0	3.0	9.3500

Apply Logistic Regression

In [20]:

```
lr = LogisticRegression()
lr.fit(X_train_trf,y_train)
y_pred = lr.predict(X_test_trf)
accuracy_score(y_test,y_pred)
```

Out[20]:

```
0.7039106145251397
```

In [ ]: