One Hot Encoding 이해하기

학습 목표

A. One Hot Encoding은 무엇인가?
B. 왜 사용할까?

A. What is One Hot Encoding?(One Hot Encoding은 무엇인가?)

가. One Hot Encoding은 머신러닝 알고리즘에서 더 나은 예측을 위해 제공되는 하나의 과정입니다.

B. Why do you need one hot encoding?

(왜 필요할까?)

Label 인코딩의 오류

Label 의 인코딩의 문제는 범주값이 높을수록 카테고리가 더 우수하다고 가정합니다.

범주형 값에 의해 가장 가치 있는 모델은
VW > Acura > Honda이다.
평균을 계산해서 확인하면 평균은 2이고, 이것이 의미하는 바는 VM와 Honda의 평균은 Acura이다.
이 내용은 오류가 발생합니다. 이 값을 가지고 모델을 예측한다는 것은 많은 오류가 있다.

LabelEncoder, OneHotEncoder

In [31]:
```
### 01. 데이터 준비
import pandas as pd
data = { "alp": ["b","c", "a", "d"],
         "Value":[2,3,8,4]
       }
df = pd.DataFrame(data)
df
```

Out[31]:

	Value	alp
0	2	b
1	3	c
2	8	a
3	4	d

In [32]:
```
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
```

In [39]:
```
## LabelEncoder
x = df.iloc[:,:].values
print(type(x), x)
```

```
<class 'numpy.ndarray'> [[2 'b']
 [3 'c']
 [8 'a']
 [4 'd']]
```

In [40]:
```
encoder_x = LabelEncoder()
x[:,1] = encoder_x.fit_transform(x[:,1])  #
x
```

Out[40]:
```
array([[2, 1],
       [3, 2],
       [8, 0],
       [4, 3]], dtype=object)
```

In [41]:
```
## OneHotEncoder
onehotencoder = OneHotEncoder(categorical_features=[1])
x = onehotencoder.fit_transform(x).toarray()
x
```

Out[41]:
```
array([[0., 1., 0., 0., 2.],
       [0., 0., 1., 0., 3.],
       [1., 0., 0., 0., 8.],
       [0., 0., 0., 1., 4.]])
```

02. LabelEncoder, OneHotEncoder 실습

In [115]:
```
data = { "companyName": ["MS","Apple", "Google", "Google"]}
df1 = pd.DataFrame(data)
df2 = df1.copy()

### OneHotEncoding
from sklearn.preprocessing import LabelEncoder, OneHotEncoder

### LabelEncoder
x = df1.iloc[:,:].values
encoder_x = LabelEncoder()
x[:,0] = encoder_x.fit_transform(x[:,0])  # df1 의 값도 변경됨.
print(x)

## OneHotEncoder
onehotencoder = OneHotEncoder(categorical_features=[0])
x = onehotencoder.fit_transform(x).toarray()
print(x)

# 변경된 값을 DataFrame형태로 변경
dx = pd.DataFrame(x, dtype=int)
```

```
[[2]
 [0]
 [1]
 [1]]
[[0. 0. 1.]
 [1. 0. 0.]
 [0. 1. 0.]
 [0. 1. 0.]]
```

In [126]:
```
print(df2)
dx
```

```
  companyName
0          MS
1       Apple
2      Google
3      Google
```

Out[126]:

	0	1	2
0	0	0	1
1	1	0	0
2	0	1	0
3	0	1	0

In [127]:
```
df2['Apple']= dx[0]
df2['Google']= dx[1]
df2['MS']= dx[2]
df2
```

Out[127]:

	companyName	Apple	Google	MS
0	MS	0	0	1
1	Apple	1	0	0
2	Google	0	1	0
3	Google	0	1	0

In [ ]:
```

```