DEALING WITH CATEGORICAL COLUMNS

By:

M. Ashish Reddy

DSWP -> Batch-5

Encoding Categorical Data

- There are three common approaches for converting ordinal and categorical variables to numerical values. They are:

1. Ordinal Encoding
2. One-Hot Encoding
3. Dummy Variable Encoding

1. Ordinal Encoding

- In ordinal encoding, each unique category value is assigned an integer value.
- For example, “*red*” is 1, “*green*” is 2, and “*blue*” is 3.
- This is called an ordinal encoding or an integer encoding and is easily reversible. Often, integer values starting at zero are used.

Input

```
# example of a ordinal encoding
from numpy import asarray
from sklearn.preprocessing import OrdinalEncoder
# define data
data = asarray([['red'], ['green'], ['blue']])
print(data)
# define ordinal encoding
encoder = OrdinalEncoder()
# transform data
result = encoder.fit_transform(data)
print(result)
```

Output

```
[['red']
 ['green']
 ['blue']]
[[2.]
 [1.]
 [0.]]
```

2. One-Hot Encoding

- For categorical variables where no ordinal relationship exists, the integer encoding may not be enough, at best, or misleading to the model at worst.
- **Input**

```
# example of a one hot encoding
from numpy import asarray
from sklearn.preprocessing import OneHotEncoder
# define data
data = asarray([['red'], ['green'], ['blue']])
print(data)
# define one hot encoding
encoder = OneHotEncoder(sparse=False)
# transform data
onehot = encoder.fit_transform(data)
print(onehot)
```

Output

```
[['red']
 ['green']
 ['blue']]
[[0. 0. 1.]
 [0. 1. 0.]
 [1. 0. 0.]]
```

3. Dummy Variable Encoding

- The one-hot encoding creates one binary variable for each category.

- **Input**

```
# example of a dummy variable encoding
from numpy import asarray
from sklearn.preprocessing import OneHotEncoder
# define data
data = asarray([['red'], ['green'], ['blue']])
print(data)
# define one hot encoding
encoder = OneHotEncoder(drop='first', sparse=False)
# transform data
onehot = encoder.fit_transform(data)
print(onehot)
```

Output

```
[['red']
 ['green']
 ['blue']]
[[0. 1.]
 [1. 0.]
 [0. 0.]]
```

When to use a Label Encoding vs. One Hot Encoding

- We apply One-Hot Encoding when:
  1. The categorical feature is **not ordinal** (like the countries above)
  2. The number of categorical features is less so one-hot encoding can be effectively applied

- We apply Label Encoding when:
  1. The categorical feature is **ordinal** (like Jr. kg, Sr. kg, Primary school, high school)
  2. The number of categories is quite large as one-hot encoding can lead to high memory consumption


Thank You!