Label Encoding in Python

In machine learning projects, we usually deal with datasets having different categorical columns where some columns have their elements in the ordinal variable category for e.g a column income level having elements as low, medium, or high in this case we can replace these elements with 1,2,3. where 1 represents '*low*' 2 '*medium*' and 3' *high*'. Through this type of encoding, we try to preserve the meaning of the element where higher weights are assigned to the elements having higher priority.

Label Encoding

Label Encoding is a technique that is used to convert categorical columns into numerical ones so that they can be fitted by machine learning models which only take numerical data. It is an important pre-processing step in a machine-learning project.

Example Of Label Encoding

Suppose we have a column *Height* in some dataset that has elements as Tall, Medium, and short. To convert this categorical column into a numerical column we will apply label encoding to this column. After applying label encoding, the Height column is converted into a numerical column having elements 0,1, and 2 where 0 is the label for tall, 1 is the label for medium, and 2 is the label for short height.

Height	Height
Tall	0
Medium	1
Short	2

Example of Label Encoding

We will apply *Label Encoding* on the iris dataset on the target column which is Species. It contains three species *Iris-setosa, Iris-versicolor, Iris-virginica*.

Python3

```
import  numpy as np
import  pandas as pd

df  =  pd.read_csv(``'../../data/Iris.csv'``)

df[``'species'``].unique()
```

Output:

```
array(['Iris-setosa', 'Iris-versicolor', 'Iris-virginica'], dtype=object)
```

After applying Label Encoding with LabelEncoder() our categorical value will replace with the numerical value[int].

Python3

```
from  sklearn  import  preprocessing

label_encoder  =  preprocessing.LabelEncoder()

df[``'species'``]``=  label_encoder.fit_transform(df[``'species'``])

df[``'species'``].unique()
```

Output:

```
array([0, 1, 2], dtype=int64)
```

Limitation of label Encoding

Label encoding converts the categorical data into numerical ones, but it assigns a unique number(starting from 0) to each class of data. This may lead to the generation of priority issues during model training of data sets. A label with a high value may be considered to have high priority than a label having a lower value.

Example For Limitation of Label Encoding

An attribute having output classes **Mexico, Paris, Dubai**. On Label Encoding, this column lets **Mexico** is replaced with *0*, **Paris** is replaced with *1*, and **Dubai** is replaced with 2.

With this, it can be interpreted that **Dubai** has high priority than **Mexico** and **Paris** while training the model, But actually, there is no such priority relation between these cities here.