Certificate Course in Machine Learning using Python [6 Weeks]

<u>Dashboard</u> My courses

Certificate Course in Machine Learning using Python [6 Weeks] Day 11

Pre-processing of data for Machine Learning

Pre-processing of data for Machine Learning Data Pre-processing for Machine Learning

Step 5: Encoding Categorical Data

- Machine learning algorithms require numerical inputs.
- Categorical data are variables that contain label values rather than numeric values.
- The number of possible values is often limited to a fixed set.
- Machine learning algorithms cannot work with variables in text form.
- · Categorical values must be transformed into numeric values to work with machine learning algorithm.

Categorical values can be transformed in to numeric values by:

- Label Encoding
- One Hot Encoding

LabelEncoder:

- Encode target labels with value between 0 and n_classes-1.
- This transformer should be used to encode target values, i.e. y, and not the input X.

Example:

```
from sklearn.preprocessing import LabelEncoder
lb_encode = LabelEncoder()
# Encode labels in column 'Country'.
X['Country'] = Ib_encode.fit_transform(X['Country'])
print(X.head())
```

Output:

```
from sklearn.preprocessing import LabelEncoder
label_encoder = LabelEncoder()

# Encode Labels in column 'Country'.
X['Country'] = label_encoder.fit_transform(X['Country'])
print(X)
```

```
Country
                          Salary
               Age
            44.000 72000.000000
0
         0
           27.000 48000.000000
1
         2
2
         1
           30.000 54000.000000
3
           42.625 52571.428571
         2
4
           40.000
                    1000.000000
           35.000 58000.000000
5
         0
6
         2
           78.000 52000.000000
7
         0 42.625 52571.428571
8
         0
            50.000 83000.000000
            37.000 52571.428571
9
```

Limitation of Label Encoding

- Label encoding convert the data in machine readable form, but it assigns a unique number(starting from 0) to each class of data.
- This may lead to the generation of priority issue in training of data sets. A label with high value may be considered to have high priority than a label having lower value.
- For example, on Label Encoding 'Country' column, let France is replaced with 0, Germany is replaced with 1 and Spain is replaced with 2.
- With this, it can be interpreted that Spain have high priority than Germany and France while training the model. But actually there is no such priority relation between these countries.
- This can be overcome by the concept of One-Hot Encoding.

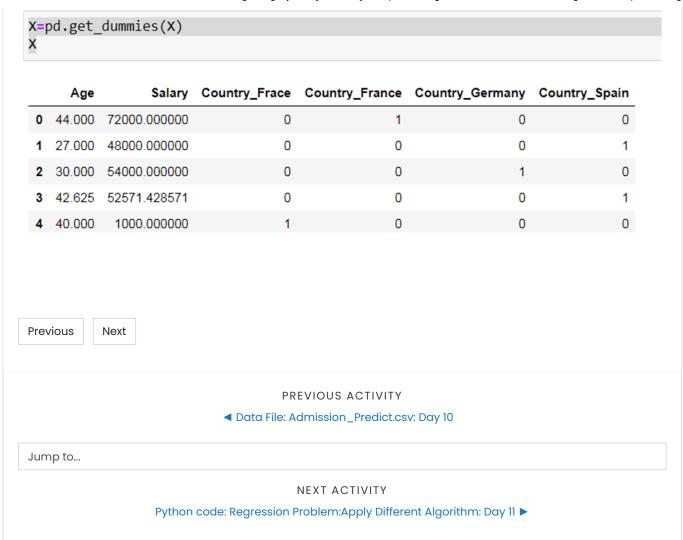
One Hot Encoding

- The technique to convert categorical values into a numerical vector is known as one hot encoding.
- It refers to splitting the column which contains numerical categorical data to many columns depending on the number of categories present in that column. Each column contains "0" or "1" corresponding to which column it has been placed.
- The resulting vector will have only one element equal to 1 and the rest will be 0.
- For example, In given dataset 'Country' column contains categorical data. So, 'Country' column must be converted into numerical values before starting Machine Learning Process.

get_dummies(): Used to encode categorical values into numerical values.

```
Syntax: get_dummies(dataframe)

Example: X=get_dummies(X)
```



Stay in touch

Contact Us

http://nielit.gov.in/gorakhpur/

<u>□ abhinav@nielit.gov.in or ajay.verma@nielit.gov.in</u>