Managing categorical data

In many classification problems, the target dataset is made up of categorical labels which cannot immediately be processed by any algorithm. An encoding is needed and scikit-learn offers at least two valid options. Let's consider a very small dataset made of 10 categorical samples with two features each:

Copy

```
import numpy as np

>>> X = np.random.uniform(0.0, 1.0, size=(10, 2))
>>> Y = np.random.choice(('Male','Female'), size=(10))
>>> X[0]
array([ 0.8236887 ,  0.11975305])
>>> Y[0]
'Male'
```

The first option is to use the `LabelEncoder` class, which adopts a dictionary-oriented approach, associating to each category label a progressive integer number, that is an index of an instance array called `classes_` :

Copy

```
from sklearn.preprocessing import LabelEncoder

>>> le = LabelEncoder()
>>> yt = le.fit_transform(Y)
>>> print(yt)
[0 0 0 1 0 1 1 0 0 1]

>>> le.classes_array(['Female', 'Male'], dtype='|S6')
```

The inverse...

❮ Previous Section (/book/big_data_and_business_intelligence/9781785889622/3/ch03lvl1s

Next Section ❯ (/book/big_data_and_business_intelligence/9781785889622/3/ch03lvl1sec2