17.6.1

Encode Labels With Pandas

It's often said that much of a data scientist's time is spent cleaning and preparing data. In machine learning, too, the data rarely comes ready for analysis.

One of the tasks involved in data preparation for machine learning is to convert textual data into numerical data.

While many datasets contain categorical features (e.g., M or F), machine learning algorithms typically only work with numerical data. Categorical and text data must therefore be converted to numerical data for use in machine learning—which is what we'll do in this section.

First, download the files you'll need for this task.

Download 17-6-1-label_encode.zip (https://2u-data-curriculum-team.s3.amazonaws.com/dataviz-online/module_17/17-6-1-label_encode.zip)

Download the files and open the Jupyter Notebook. We'll first import the modules we'll use and open the dataset in a Pandas DataFrame with the following code:

```
import pandas as pd
from path import Path

file_path = Path("../Resources/loans_data.csv")
loans_df = pd.read_csv(file_path)
loans_df.head()
```

A preview of the DataFrame reveals seven columns: six features and a target. The dataset contains simulated loan data. There are 500 records, and each row represents a loan application:

	amount	term	month	age	education	gender	bad
0	1000	30	June	45	High School or Below	male	0
1	1000	30	July	50	Bachelor	female	0
2	1000	30	August	33	Bachelor	female	0
3	1000	15	September	27	college	male	0
4	1000	30	October	28	college	female	0

The dataset includes the following columns:

- **Amount:** The loan amount in U.S. dollars.
- **Term:** The loan term in months.
- **Month:** The month of the year when the loan was requested.
- **Age:** Age of the loan applicant.
- **Education:** Educational level of the loan applicant.
- **Gender:** The sex of the loan applicant.
- **Bad:** Status of the application (1: bad, or denial; 0: good, or approval).

IMPORTANT

Scikit-learn's algorithms only understand numeric data.

To use Scikit-learn's machine learning algorithms, the text features (month, education, and gender) will have to be converted into numbers. This process is called encoding. Furthermore, the steps taken to prepare the data to make them usable for building machine learning models are called preprocessing. Encoding text labels into numerical values is one preprocessing step. Later we'll discuss scaling, another preprocessing step.

The first and the simplest encoding we'll perform in this dataset is with the `gender` column, which contains only two values: male and female. We'll convert these values into numerical ones with the `pd.get_dummies()` method:

```
loans_binary_encoded = pd.get_dummies(loans_df, columns=["gender"])
loans_binary_encoded.head()
```

The method takes two arguments:

- The first argument for `pd.get_dummies()` here is the DataFrame.
- The second argument specifies the column to be encoded.

	amount	term	month	age	education	bad	gender_female	gender_male
0	1000	30	June	45	High School or Below	0	0	1
1	1000	30	July	50	Bachelor	0	1	0
2	1000	30	August	33	Bachelor	0	1	0
3	1000	15	September	27	college	0	0	1
4	1000	30	October	28	college	0	1	0

The gender column has split into two columns, `gender_female` and `gender_male`, with each column now containing 0 (false) or 1 (true). Since the first row represents a male loan applicant, the `gender_female` column reads 0 and the `gender_male` column reads 1.

It's also possible to encode multiple columns at the same time.

```
loans_binary_encoded = pd.get_dummies(loans_df, columns=["education", "gender"])
loans_binary_encoded.head()
```

	amount	term	month	age	bad	education_Bachelor	education_High School or Below	education_Master or Above	education_college	gender_female	gender_male
0	1000	30	June	45	0	0	1	0	0	0	1
1	1000	30	July	50	0	1	0	0	0	1	0
2	1000	30	August	33	0	1	0	0	0	1	0
3	1000	15	September	27	0	0	0	0	1	0	1
4	1000	30	October	28	0	0	0	0	1	1	0

As before, the `gender` column has split into two columns. The education column has split into four columns (`Bachelor, High School or Below`, `Master or Above`, and `college`), with an associated 0 or 1. If a loan applicant has a bachelor's degree, that column will read 1, and the others (`High School or Below`, `Master or Above`, and `college`) will read 0. For an applicant who did not graduate from high school, the `education_Bachelor`, `education_Master or Above`, and `education_college` columns will be 0, and the `education_High School or Below` will show 1.