Resumen del capítulo: Lectura y visualización de datos

Solucionar problemas con archivos CSV

Recuerda que CSV significa **valores separados por comas**. Sin embargo, un archivo CSV no tiene que usar solo una coma como **delimitador**; se puede usar cualquier carácter. Por ejemplo, los valores separados por tabuladores son otro formato común.

Puedes cambiar el delimitador utilizando el parámetro `sep=`. También puedes establecer los nombres de las columnas y crear un encabezado con los parámetros `names=` y `header=`:

```
import pandas as pd

column_names = [
    'country',
    'name',
    'capacity_mw',
    'latitude',
    'longitude',
    'primary_fuel',
    'owner'
]
data = pd.read_csv('/datasets/gpp_modified.csv', sep='|', header=None, names=column_names)

print(data.head())
```

Subir diferentes hojas de archivos Excel

Los archivos de Excel pueden constar de varias hojas. Al usar la función `read_excel()`, importas solo la primera hoja de forma predeterminada. Si necesitas trabajar con la otra hoja, utiliza el parámetro `sheet_name=`:

```
import pandas as pd

df = pd.read_excel('/datasets/product_reviews.xlsx', sheet_name='reviewers')

print(df.head())
```

Observación de los datos

Nunca es mala idea llamar al método `info()` cada vez que empieces a trabajar con un nuevo DataFrame. El método `info()` no devuelve nada, sino que imprime información general sobre el DataFrame. Pero para obtener una visión aún más completa de tus datos, también puede ser necesario usar el método `sample()` que selecciona filas aleatorias del DataFrame:

```
import pandas as pd

column_names = [
    'country',
    'name',
    'capacity_mw',
    'latitude',
    'longitude',
    'primary_fuel',
    'owner'
]
data = pd.read_csv(
    '/datasets/gpp_modified.csv',
    sep='|',
    header=None,
    names=column_names,
    decimal=',',
)

print(data.sample(5))
```

Descripciones numéricas y describe()

Y una comprensión aún más avanzada de los datos se puede obtener con el método `describe()`, que muestra las principales medidas estadísticas del conjunto de datos, incluida la desviación estándar, los valores mínimos y máximos, cuando se solicitan valores numéricos:

```
print(data.describe())
```

```
        capacity_mw      latitude     longitude
count  34936.000000  34936.000000  34936.000000
mean     163.355148     32.816637     -6.972803
```

```
std      489.636072     22.638603     78.405850
min        1.000000    -77.847000   -179.977700
25%        4.900000     29.256475    -77.641550
50%       16.745000     39.727750     -2.127100
75%       75.344250     46.263125     49.502675
max    22500.000000     71.292000    179.388700
```

Sin embargo, para los valores categóricos, el método `describe()` produce un efecto ligeramente diferente, centrándose principalmente en la cantidad de valores únicos y los más frecuentes:

```
print(data['country'].describe())
```

```
count                       34936
unique                        167
top       United States of America
freq                         9833
Name: country, dtype: object
```