Học phần Học Máy: Bài thực hành số 1

Phạm Tiến Lâm, Đặng Văn Báu

1. Ôn tập Python cơ bản

Activity 1. Thực hiện lại bài Code Introduction (đã up lên Google Classroom)

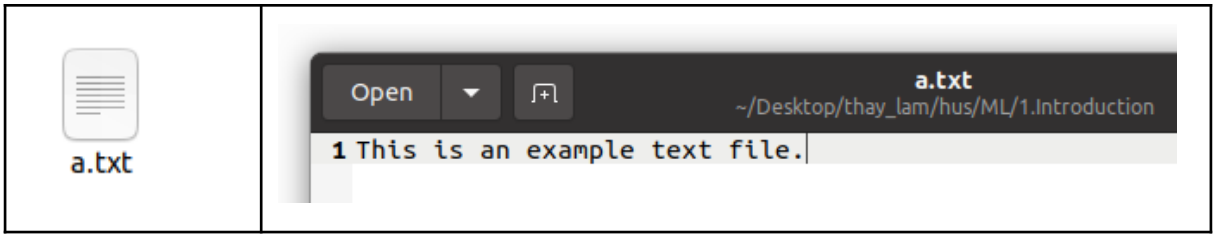
Activity 2. Các thao tác xử lý tệp văn bản (text file) với Python

Python cung cấp rất nhiều cách để đọc và ghi tệp văn bản (text file). Dưới đây là một số thao tác xử lý tệp văn bản phổ biến với Python:

Đọc tệp văn bản:

Để đọc tệp văn bản, ta sử dụng hàm open() với chế độ đọc ("r") và sau đó đọc từng dòng trong tệp sử dụng vòng lặp for.

Ví dụ: Sử dụng một file 'a.txt' đã có trước với nội dung như sau:



- Sử dụng Python đọc file 'r' :

```
1 with open("a.txt", "r") as f:
2     for line in f:
3         print(line)
This is an example text file.
```

Ghi vào tệp văn bản

Để ghi vào tệp văn bản, ta sử dụng hàm open() với chế độ ghi ("w") hoặc chế độ ghi thêm ("a"). Sau đó, ta ghi nội dung vào tệp bằng cách sử dụng phương thức write().

- Ghi vào tệp ('w'):

```
with open("b.txt", "w") as f:
    f.write("Hello world!")

with open("b.txt", "r") as f:
    contents = f.read()

print(contents)
```

```
Hello world!
```

- Ghi thêm vào cuối tệp ('a'):

```
with open("b.txt", "a") as f:
    f.write("\nHello again!")

with open("b.txt", "r") as f:
    contents = f.read()

print(contents)
```

```
Hello world!
Hello again!
```

Xóa tệp văn bản

Để xóa tệp văn bản, ta sử dụng hàm os.remove() và truyền đường dẫn đến tệp văn bản cần xóa

```
import os

os.remove("b.txt")

```

Trong Python, ngoài các chế độ mở tệp văn bản "r" (đọc), "w" (ghi), và "a" (ghi thêm), còn có các chế độ mở tệp "r+" (đọc và ghi), "w+" (ghi và đọc) và "a+" (ghi thêm và đọc).

Các chế độ mở tệp văn bản đó có chức năng như sau:

- "r+": Mở tệp văn bản để đọc và ghi. Tệp văn bản được mở ở vị trí đầu tiên của tệp. Nếu tệp không tồn tại, thì sẽ có lỗi.
- w+ : Mở tệp cho cả ghi và đọc. Ghi đè lên tệp hiện có nếu tệp tồn tại. Nếu tệp không tồn tại, hãy tạo một tệp mới để đọc và ghi.
- "a+": Mở tệp văn bản để ghi thêm và đọc. Tệp văn bản được mở ở vị trí cuối cùng của tệp. Nếu tệp không tồn tại, tệp sẽ được tạo mới.

Để sử dụng các chế độ mở tệp văn bản này, ta cũng sử dụng hàm open():

```python
# Mở tệp văn bản để ghi và đọc
with open("b.txt", "w+") as f:
    # Ghi nội dung vào tệp
    f.write("Hello world")

    # Đóng tệp
    f.close()
```

```python
with open("b.txt", "a+") as f:
    # Thêm nội dung vào cuối tệp
    f.write("Hello again!")

    # Đóng tệp
    f.close()
```

```python
# Mở tệp văn bản để đọc và ghi
with open("a.txt", "r+") as f:
    # Đọc nội dung tệp
    contents = f.read()

    # Thêm nội dung vào đầu tệp
    f.write("\n")
    f.write("New content")

    # Đóng tệp
    f.close()
```

Activity 3. Thao tác với file ".csv" sử dụng thư viện Pandas:
(link download data: https://www.kaggle.com/datasets/rakeshrau/social-network-ads)

A. Đọc file và trích xuất thông tin từ dữ liệu:

- Khai báo thư viện Pandas:

```
1 import pandas as pd
```

- Đọc file ".csv"

```
1 data = pd.read_csv("/home/bau/Downloads/Social_Network_Ads.csv")
2
3 data.head(5)
```

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0

- Hiển thị tên các cột:

```
1 print(data.columns)
```

```
Index(['User ID', 'Gender', 'Age', 'EstimatedSalary', 'Purchased'], dtype='object')
```

- Hiển thị kích thước bảng:

```
1 data.shape
```

```
(400, 5)
```

- Trích xuất thông tin trong cột "Age":

```
1 age_col = data.loc[:, ['Age']].values
2 age_col
```

```
array([[19],
       [35],
       [26],
       [27],
       [19],
```

```
1 age_col = data.iloc[:, [2]].values
2 age_col
```

```
array([[19],
       [35],
       [26],
       [27],
       [19],
```

B. Xử lý dữ liệu:

- Phương thức isnull() và notnull() để kiểm tra phần tử là NaN hay không là NaN

```
1 data.notnull()
```

	User ID	Gender	Age	EstimatedSalary	Purchased
0	True	True	True	True	True
1	True	True	True	True	True
2	True	True	True	True	True
3	True	True	True	True	True
4	True	True	True	True	True
...	...	...	...	...	...
395	True	True	True	True	True
396	True	True	True	True	True
397	True	True	True	True	True
398	True	True	True	True	True
399	True	True	True	True	True

400 rows × 5 columns

```
1 data.isnull()
```

	User ID	Gender	Age	EstimatedSalary	Purchased
0	False	False	False	False	False
1	False	False	False	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	False	False	False
...	...	...	...	...	...
395	False	False	False	False	False
396	False	False	False	False	False
397	False	False	False	False	False
398	False	False	False	False	False
399	False	False	False	False	False

400 rows × 5 columns

- Phương thức kiểm tra hàng/cột chứa phần tử null:

```
In [91]: 1 data.isnull().any()
Out[91]: User ID            False
         Gender             False
         Age                False
         EstimatedSalary    False
         Purchased          False
```

```
In [92]: 1 data.isnull().any(1)
Out[92]: 0    False
         1    False
         2    False
         3    False
         4    False
              ...
```

- Drop các hàng chứa phần tử là "NaN":

```
In [89]: 1 print('Drop các hàng có chứa phần tử là NaN')
         2 data.dropna()
Drop các hàng có chứa phần tử là NaN
```

Out[89]:

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0
...	...	...	...	...	...

- Thay thế các giá trị NaN thành 1 giá trị mới:

```
print('Gán gián trị mặc định cho "missing data" dùng hàm fillna(number)')
data.fillna(10000000)
```

```
Gán gián trị mặc định cho "missing data" dùng hàm fillna(number)
```

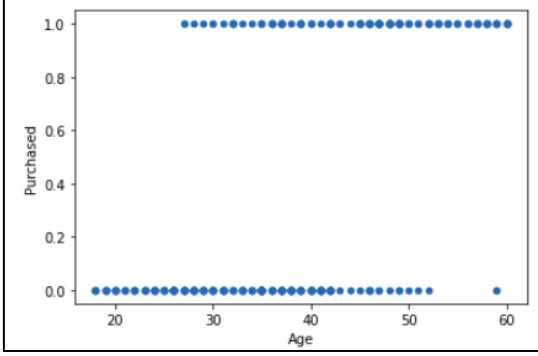
	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0
...	...	...	...	...	...

C. Trực quan hóa dữ liệu:

- Sử dụng biểu đồ "Scatter" biểu diễn sự tương quan giữa tuổi và mua/không mua:

```
data.plot(kind = 'scatter', x = 'Age', y = 'Purchased')

plt.show()
```



- Sử dụng biểu đồ "Histogram" biểu diễn phân bố tuổi của các khách hàng trong bảng dữ liệu:

```
data["Age"].plot(kind = 'hist')
```

```
<AxesSubplot:ylabel='Frequency'>
```