

Pandas

CSCI3329

Pandas

Pandas is a python external engineering library.

It puts the data in a tabular **data frame** so that it can be used easily, in simple terms, like Excel in Python.

```
import pandas as pd
```

Why not Excel?

It is also possible with Excel, but it is very slow when processing large volumes.

And when you have to do the same thing over and over again, coding in Pandas is much more productive.

For machine learning, **data preprocessing** is used for 70% of the entire process, and the rest is used for machine learning. When receiving data into Excel, refining and splitting it, it may take less manpower or time to do it by a machine than by a human.

**Machine Learning is the study of computer algorithms that improve automatically through experience. It is seen as a subset of artificial intelligence. Machine learning algorithms build a model based on sample data, known as "training data", in order to make predictions or decisions without being explicitly programmed to do so.

Reading a file and dataframe

Prepare a CSV file like the example below. There are 7 cars and the file name is `cars.csv`.

	A	B	C
1	Brand	Model	Year
2	Toyota	Camry	2021
3	Honda	Accord	2020
4	Ford	F150	2018
5	Nissan	Altima	2015
6	BMW	330	2019
7	Audi	A4	2020
8	Hyundai	Sonata	2017

Reading a file and dataframe

Read the file using `read_csv` function that returns a dataframe.

```
1 import pandas as pd
2
3 dataframe = pd.read_csv("cars.csv")
4 print(dataframe)
```

```
/home/dkim/PycharmProjects/CSCI3328/venv/bin/python /home/dkim/Py
```

	Brand	Model	Year
0	Toyota	Camry	2021
1	Honda	Accord	2020
2	Ford	F150	2018
3	Nissan	Altima	2015
4	BMW	330	2019
5	Audi	A4	2020
6	Hyundai	Sonata	2017

```
Process finished with exit code 0
```

dataframe

A Pandas' dataframe consists of Series. Each column is a **Series** object which is actually a list.

```
1 import pandas as pd
2
3 dataframe = pd.read_csv("cars.csv")
4 print(type(dataframe.Model))
```

```
/home/dkim/PycharmProjects/CSCI3328/venv/bin/python /home/dkim/PycharmPro
<class 'pandas.core.series.Series'>
```

```
Process finished with exit code 0
```

dataframe

A Pandas' dataframe consists of Series. Each column is a **Series** object which is actually a list.

```
1 import pandas as pd
2
3 n = pd.core.series.Series(['Kim', 'Lee', 'Park'])
4 a = pd.core.series.Series([35, 24, 44])
5
6 dataframe = pd.DataFrame(data=dict(name=n, age=a))
7 print(dataframe)
```

```
/home/dkim/PycharmProjects/CSCI3328/venv/bin/python /h
```

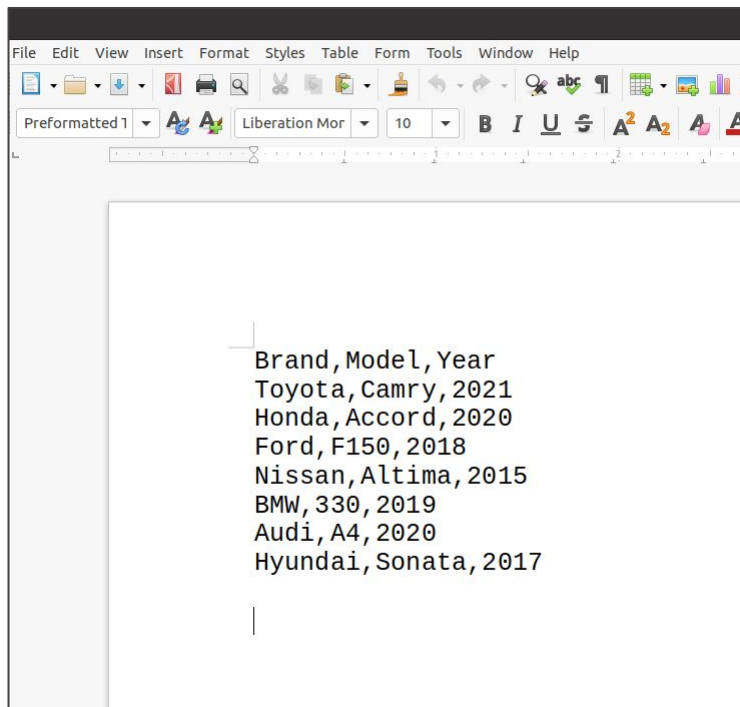
```
    name  age  
0    Kim   35  
1    Lee   24  
2   Park   44
```

```
Process finished with exit code 0
```


How about .txt file?

If the `txt` file has the data format as a `csv` (comma separated value), you can use `read_csv` function for the `txt` file as well.

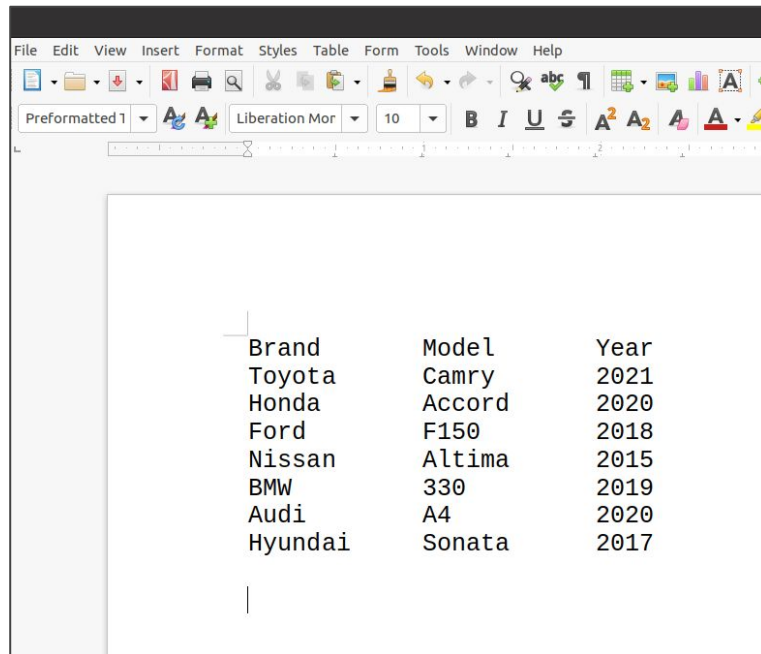
```
1 import pandas as pd
2
3 dataframe = pd.read_csv("cars.txt")
4 print(dataframe)
```



How about non-csv text file?

If your data is not a csv file and the delimiter is not a comma, you can specify what delimiter you want to use. For example,

```
1 import pandas as pd
2
3 dataframe = pd.read_csv("cars_tab.txt", delimiter='\t')
4 print(dataframe)
```



A screenshot of a word processor window showing a table of car data. The table has three columns: Brand, Model, and Year. The data is as follows:

Brand	Model	Year
Toyota	Camry	2021
Honda	Accord	2020
Ford	F150	2018
Nissan	Altima	2015
BMW	330	2019
Audi	A4	2020
Hyundai	Sonata	2017

Header name

What if the data you have does not have a header (column names)?

```
1 import pandas as pd
2
3 dataframe = pd.read_csv("cars_no_head.csv")
4 print(dataframe)
```

```
/home/dkim/PycharmProjects/CSCI3328/venv/b
```

```
   Toyota  Camry  2021
0   Honda  Accord  2020
1   Ford   F150   2018
2   Nissan  Altima  2015
3     BMW    330   2019
4   Audi    A4    2020
5  Hyundai  Sonata  2017
```

```
Process finished with exit code 0
```

```
1 Toyota,Camry,2021
2 Honda,Accord,2020
3 Ford,F150,2018
4 Nissan,Altima,2015
5 BMW,330,2019
6 Audi,A4,2020
7 Hyundai,Sonata,2017
```

cars_no_head.csv

Adding a header

Method 1

```
1 import pandas as pd
2
3 dataframe = pd.read_csv("cars_no_head.csv")
4 dataframe.columns = ['Brand', 'Model', 'Year']
5 print(dataframe)
```

Adding a header

Method 2

```
1 import pandas as pd
2
3 dataframe = pd.read_csv("cars_no_head.csv", header=None, names=['Brand', 'Model', 'Year'])
4 print(dataframe)
```

Creating a dataframe

Using a Python `list`

```
1 import pandas as pd
2
3 mydict = [['Toyota', 'Sienna', 2018],
4           ['Honda', 'Civic', 2004],
5           ['Audi', 'A6', 2009]]
6
7 column_names = ['brand', 'model', 'year']
8 df = pd.DataFrame(mydict, columns=column_names)
9 print(df)
```

```
/home/dkim/PycharmProjects/CSCI3328,
   brand  model  year
0  Toyota  Sienna  2018
1   Honda   Civic  2004
2   Audi     A6   2009
```

Creating a dataframe

Using a Python dictionary

```
1 import pandas as pd
2
3 mydict = [{'brand': 'Toyota', 'model': 'Camry', 'year': 2021},
4           {'brand': 'Honda', 'model': 'Accord', 'year': 2020},
5           {'brand': 'Ford', 'model': 'F150', 'year': 2019}]
6
7 df = pd.DataFrame(mydict)
8 print(df)
```

```
/home/dkim/PycharmProjects/CSCI3328/venv/
  brand  model  year
0 Toyota  Camry  2021
1 Honda   Accord 2020
2 Ford    F150   2019

Process finished with exit code 0
```

Creating a CSV file

```
1  import pandas as pd
2
3  mydict = [['Toyota', 'Sienna', 2018],
4           ['Honda', 'Civic', 2004],
5           ['Audi', 'A6', 2009]]
6
7  column_names = ['brand', 'model', 'year']
8  df = pd.DataFrame(mydict, columns=column_names)
9  print(df)
10
11 df.to_csv('car.csv', header=False, index=False)
```


Lab 30-1

Make a python program that creates a CSV file using Pandas (`to_csv()` function). The data should have at least three columns (e.g., `name`, `age`, `job`) and five rows

After creating the CSV file, then read the CSV file using the function `read_csv()` and display the `Dataframe` on the console.

Submit a python file, csv file, and screenshot of the output.

slicing

Select top five rows

`Dataframe.head(n)` displays top `n` rows (default is 5).

```
1  import pandas as pd
2
3  mydict = [['Toyota', 'Sienna', 2018],
4            ['Honda', 'Civic', 2004],
5            ['Ford', 'F250', 2013],
6            ['Hyundai', 'Sonata', 2020],
7            ['Honda', 'Accord', 2021],
8            ['BMW', '330i', 2020],
9            ['Mercedes', 'C300', 2020],
10           ['Audi', 'A6', 2009]]
11
12  column_names = ['brand', 'model', 'year']
13  df = pd.DataFrame(mydict, columns=column_names)
14  print(df.head())
```

Select rows by index

`Dataframe[i:j]` selects rows which index is `i` through `j-1`

```
1 import pandas as pd
2
3 mydict = [['Toyota', 'Sienna', 2018],
4           ['Honda', 'Civic', 2004],
5           ['Ford', 'F250', 2013],
6           ['Hyundai', 'Sonata', 2020],
7           ['Honda', 'Accord', 2021],
8           ['BMW', '330i', 2020],
9           ['Mercedes', 'C300', 2020],
10          ['Audi', 'A6', 2009]]
11
12 column_names = ['brand', 'model', 'year']
13 df = pd.DataFrame(mydict, columns=column_names)
14 print(df[1:3])
```

```
/home/dkim/PycharmProjects/CSCI3328/venv/bin/python
brand model year
1 Honda Civic 2004
2 Ford F250 2013

Process finished with exit code 0
```

Select by a condition of column values

For example, `dataframe[dataframe.year > 2019]` displays rows which year is greater than 2019.

```
1 import pandas as pd
2
3 mydict = [['Toyota', 'Sienna', 2018],
4           ['Honda', 'Civic', 2004],
5           ['Ford', 'F250', 2013],
6           ['Hyundai', 'Sonata', 2020],
7           ['Honda', 'Accord', 2021],
8           ['BMW', '330i', 2020],
9           ['Mercedes', 'C300', 2020],
10          ['Audi', 'A6', 2009]]
11
12 column_names = ['brand', 'model', 'year']
13 df = pd.DataFrame(mydict, columns=column_names)
14 print(df[df.year > 2019])
```

```
/home/dkim/PycharmProjects/CSCI3328/venv/bi
      brand  model  year
3  Hyundai  Sonata  2020
4   Honda  Accord  2021
5    BMW    330i  2020
6 Mercedes   C300  2020

Process finished with exit code 0
```

Select columns by name

```
print(df[['brand', 'model']])
```

	brand	model
0	Toyota	Sienna
1	Honda	Civic
2	Ford	F250
3	Hyundai	Sonata
4	Honda	Accord
5	BMW	330i
6	Mercedes	C300
7	Audi	A6

```
Process finished with exit code 0
```

Drop rows and columns

```
1 import pandas as pd
2
3 mydict = [['Toyota', 'Sienna', 2018],
4           ['Honda', 'Civic', 2004],
5           ['Ford', 'F250', 2013],
6           ['Hyundai', 'Sonata', 2020],
7           ['Honda', 'Accord', 2021],
8           ['BMW', '330i', 2020],
9           ['Mercedes', 'C300', 2020],
10          ['Audi', 'A6', 2009]]
11
12 column_names = ['brand', 'model', 'year']
13 df = pd.DataFrame(mydict, columns=column_names)
14 #df = df.drop(df.index[1:3])
15 df = df.drop([1, 2])
16 print(df)
```

	brand	model	year
0	Toyota	Sienna	2018
3	Hyundai	Sonata	2020
4	Honda	Accord	2021
5	BMW	330i	2020
6	Mercedes	C300	2020
7	Audi	A6	2009

Process finished with exit code 0

Drop rows and columns

```
1 import pandas as pd
2
3 mydict = [['Toyota', 'Sienna', 2018],
4           ['Honda', 'Civic', 2004],
5           ['Ford', 'F250', 2013],
6           ['Hyundai', 'Sonata', 2020],
7           ['Honda', 'Accord', 2021],
8           ['BMW', '330i', 2020],
9           ['Mercedes', 'C300', 2020],
10          ['Audi', 'A6', 2009]]
11
12 column_names = ['brand', 'model', 'year']
13 df = pd.DataFrame(mydict, columns=column_names)
14 #df = df.drop(df.index[1:3])
15 #df = df.drop([1, 2])
16 df = df.drop(columns=['year'])
17 print(df)
```

```
   brand  model
0  Toyota  Sienna
1   Honda  Civic
2    Ford  F250
3 Hyundai  Sonata
4   Honda  Accord
5    BMW   330i
6 Mercedes  C300
7    Audi   A6
```

Process finished with exit code 0

Lab 30-2

Make a CSV file that has five persons' profile (name, age, job).

Read the CSV file and create a dataframe for the data.

Remove the rows which age is less than 20 from the dataframe, then save the dataframe as a CSV file.

Submit a python file and csv file