Department of Computer Science and Engineering

Course Code:CSE422 **Credits: 1.5**

Course Name: Artificial Intelligence **Prerequisite:** CSE111, CSE221

Lab 03

Data preprocessing for Artificial Intelligence (Learning Pandas)

I. Lab Overview:

Learn Python programming using Pandas library for data preprocessing. Pandas is one of the most popular Python libraries for Data Analytics. It's the "SQL of Python."

II. Why Python for AI course:

AI (artificial intelligence) opens up a world of possibilities. By taking advantage of machine learning or deep learning, you could produce many fascinating applications. But, which programming language should you use? You want a language having wide range of well documented libraries and a large community of programmers. Hence, whatever you want to do can be found in web as a reference. Python has all these advantages.

III. Lesson Fit:

There is pre-requisite to this lab: CSE111, CSE221. You should have intensive Programming Knowledge and capability of understand algorithms.

IV. Acceptance and Evaluation

Performed lab tasks will be evaluated by the Lab Instructor (LI)

a. Short viva will be conducted in each Lab or occasionally to examine your work.
b. You may work in groups but be aware that you will be evaluated individually; hence active participation during the Lab work demonstration is recommended.
c. There will be Lab handout after your work you have to handover it to LI

Lab 03

V. Learning Outcome:

After this Lab, the students will be able to:

a. Understand basic python codes
b. Get an overview how to use python using pandas for data preprocessing

VI. Activity Detail

a. Hour: 1

Getting Started:

I. Have a glance at Books "Python code for Artificial Intelligence: Foundations of Computational Agents," by David L. Poole and Alan K. Mackworth, May 28, 2018
II. "Artificial Intelligence with Python written by Prateek Joshi, January 2017
III. Check \\TSR to see e-book copy and codes, tutorials and useful links

VII. Pandas Reading Data Files, DataFrames, Data Selection

You might have your data in .csv files or Excel files Or something else. If you want to analyze that data using pandas, the first step will be to read it into a data structure that's compatible with pandas.

VIII. Pandas data structures

There are two types of data structures in pandas: **Series** and **DataFrames**.

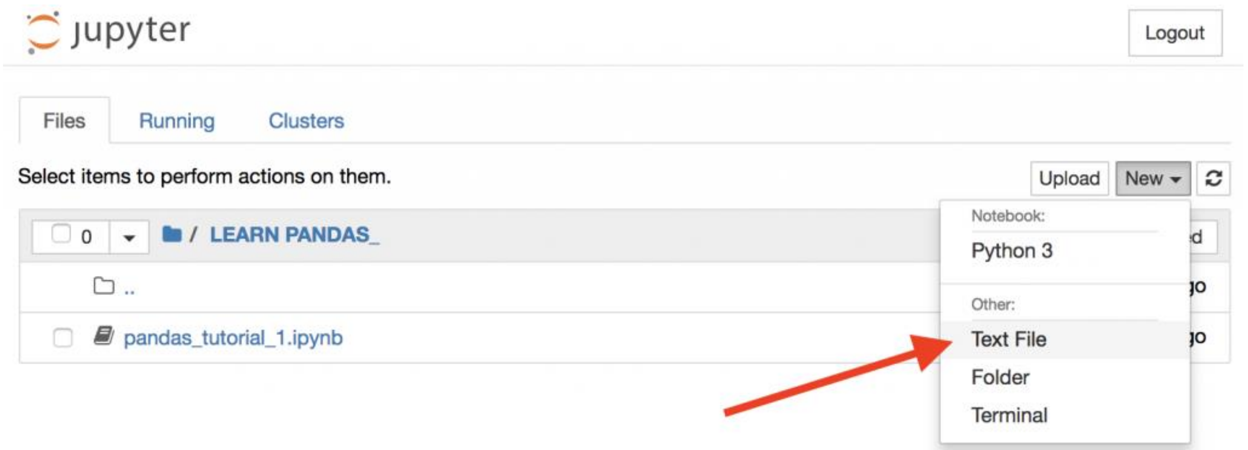
Series: a pandas Series is a one dimensional data structure (*"a one dimensional ndarray"*) that can store values — and for every value it holds a unique index, too.

DataFrame: a pandas DataFrame is a two (or more) dimensional data structure – basically a table with rows and columns. The columns have names and the rows have indexes.

IX. Loading a .csv file into a pandas DataFrame

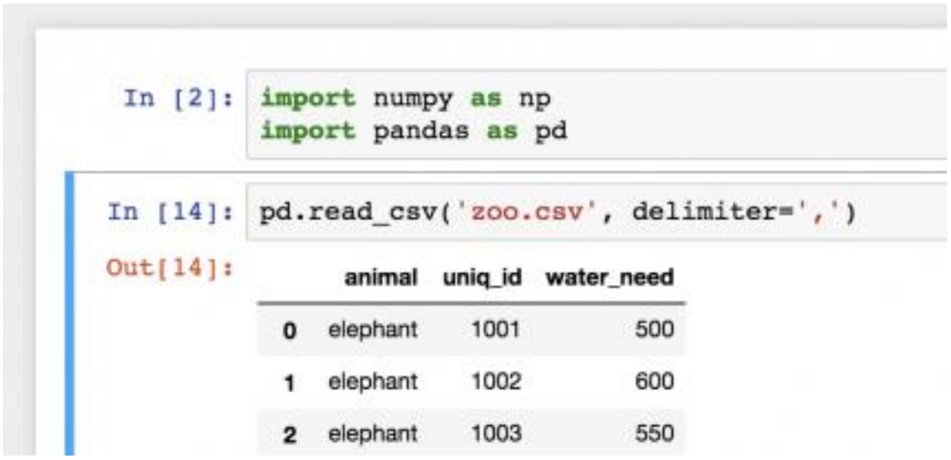
Lab 03

Go back to your Jupyter Home tab and create a new text file…



Copy and paste all data from “zoo.csv” into this file and save it as “zoo.csv” file

pd.read_csv('zoo.csv', delimiter = ',')



```
In [2]: import numpy as np
        import pandas as pd

In [14]: pd.read_csv('zoo.csv', delimiter=',')
Out[14]:
```

	animal	uniq_id	water_need
0	elephant	1001	500
1	elephant	1002	600
2	elephant	1003	550

Your data can be semicolon separated:

pd.read_csv('pandas_tutorial_read.csv', delimiter=';')

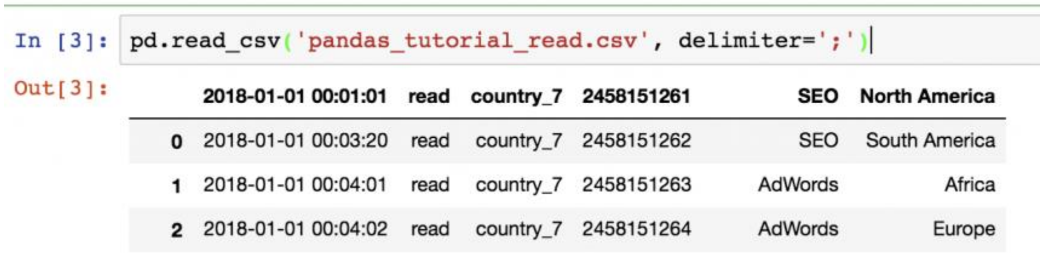
Without separator

reviews = pd.read_csv(ign.csv)

# Print out first 4 observations

print(reviews [0:4])

print(reviews.iloc[2])

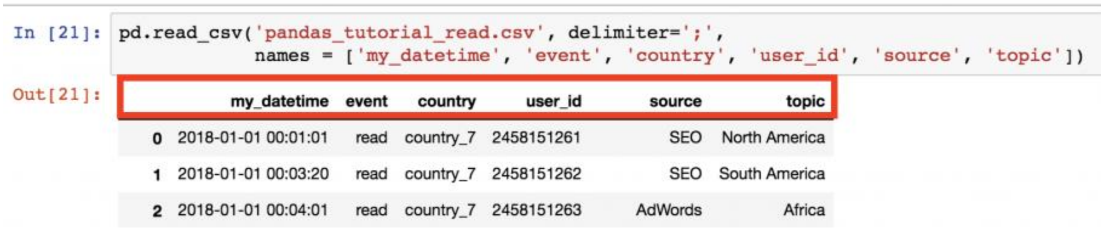


```
In [3]: pd.read_csv('pandas_tutorial_read.csv', delimiter=';')
Out[3]:
```

	2018-01-01 00:01:01	read	country_7	2458151261	SEO	North America
0	2018-01-01 00:03:20	read	country_7	2458151262	SEO	South America
1	2018-01-01 00:04:01	read	country_7	2458151263	AdWords	Africa
2	2018-01-01 00:04:02	read	country_7	2458151264	AdWords	Europe

Did you notice something? Yes, this time we didn't have a header in our csv file, so we have to set it up manually! Add the names parameter to your function!

pd.read_csv('pandas_tutorial_read.csv', delimiter=';', names = ['my_datetime', 'event', 'country',



```
In [21]: pd.read_csv('pandas_tutorial_read.csv', delimiter=';',
                     names = ['my_datetime', 'event', 'country', 'user_id', 'source', 'topic'])
Out[21]:
```

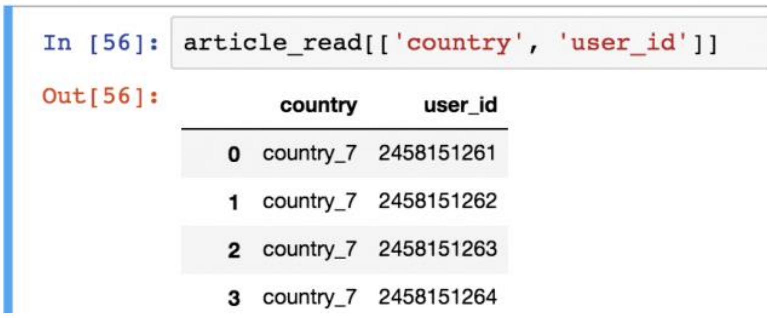
	my_datetime	event	country	user_id	source	topic
0	2018-01-01 00:01:01	read	country_7	2458151261	SEO	North America
1	2018-01-01 00:03:20	read	country_7	2458151262	SEO	South America
2	2018-01-01 00:04:01	read	country_7	2458151263	AdWords	Africa

X. Print a sample of your dataframe

```
article_read.head()
article_read.tail()
article_read.sample(5)
```

Select specific columns of your dataframe

article_read[['country', 'user_id']]



```
In [56]: article_read[['country', 'user_id']]
Out[56]:
```

	country	user_id
0	country_7	2458151261
1	country_7	2458151262
2	country_7	2458151263
3	country_7	2458151264

You can get a Series using any of these two syntaxes (and selecting only one column). Output is a Series object and not a DataFrame object

article_read.user_id

article_read['user_id']

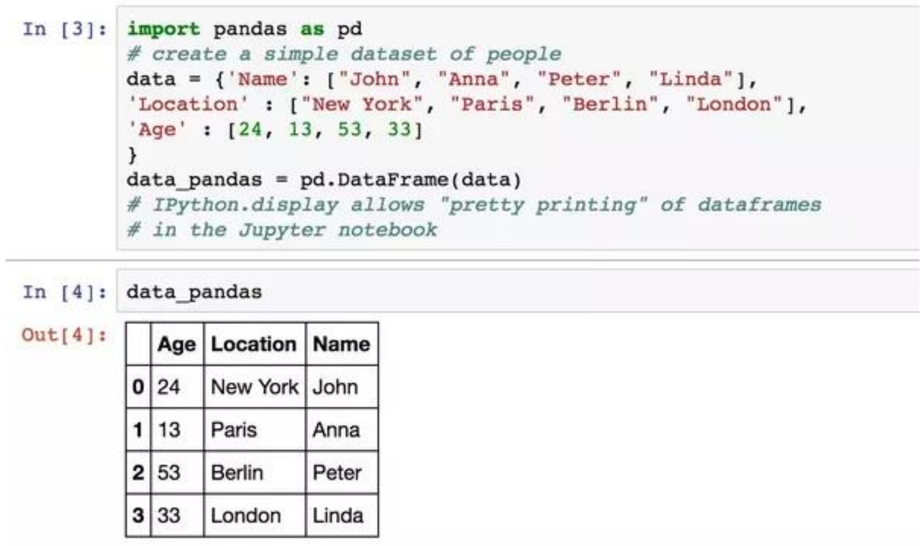
XI. Filter for specific values in your dataframe

If the previous one was a *bit* tricky, this one will be *really* tricky!

Let's say, you want to see a list of only the users who came from the 'SEO' source. In this case you have to filter for the 'SEO' value in the 'source' column:

article_read[article_read.source == 'SEO']

It's worth it to understand how pandas thinks about data filtering:



```
In [3]: import pandas as pd
        # create a simple dataset of people
        data = {'Name': ["John", "Anna", "Peter", "Linda"],
        'Location' : ["New York", "Paris", "Berlin", "London"],
        'Age' : [24, 13, 53, 33]
        }
        data_pandas = pd.DataFrame(data)
        # IPython.display allows "pretty printing" of dataframes
        # in the Jupyter notebook

In [4]: data_pandas
Out[4]:
```

	Age	Location	Name
0	24	New York	John
1	13	Paris	Anna
2	53	Berlin	Peter
3	33	London	Linda

a. **Hour: 2-3**

   You have to complete the task assigned in activities List

   I. Learning Pandas Library and built in functions

   II. Complete the given assigned task

Lab 03

Activity List

Task 01: Mark 10 **Time:** 1 Hour

Write a Python program that will create a dataframe from the given data dictionary below, then print the “attempts” column from created dataframe. After that convert the qualified column values into 0’s and 1’s, 1 for Yes and 0 for No values.

examinee = {name: [Anastasia, Dima, Katherine, James, Emily, Michael, Matthew, Laura, Kevin, Jonas],
scores: [12.5, 9, 16.5, 2.3, 9, 20, 14.5, 4.5, 8, 19],
attempts: [1, 3, 2, 3, 2, 3, 1, 1, 2, 1],
qualified: [yes, no, yes, no, no, yes, yes, no, no, yes]}

Evaluation Process (VIVA): You have to explain your program to the Lab Instructor

Hints: You have to use dataframe.map or dataframe.replace function to do this task. You can google it to see appropriate examples for “map” or “replace” function.

Evaluation Process (VIVA): You have to explain your program to the Lab Instructor

Task 02: Mark 10 **Time:** 1 Hour

Write a program using Pandas library to

1) read a csv file called “StockPriceData.csv” given in TSR. Create a dataframe using only “Adj Close” column/feature from dataset.

   Hints: use ‘usecols= ['Adj Close']’ keyword in the ‘read_csv’ method.
2) Create a chart using ‘.plot()‘ command and visualize your dataset.
3) Since you will see a noisy signal you can use rolling window method to get a smoother look. Use dataframe.rolling(100).mean().plot(). Here you can change the window size to get your desired result.
4) Let’s assign a variable for smoothing signal and save it within your dataframe
   ‘MySmoothe= dataframe.rolling(100).mean()’
   MyDataframe[‘mynewcolumn’] = MySmoothe
5) Show your dataset first 20 rows using ‘head(20)’ command
6) Drop the ‘nan’ value using ‘YourDataframe.dropna()’ command
7) Create a chart using plot() command again, you will see sometimes smoothed signal line intersecting the noisy signal line and vice versa.

8) Use different values for rolling().mean() function. So, roughly which values are giving better prediction to determine uptrend or downtrend? Why?

Evaluation Process (VIVA): You have to explain your program to the Lab Instructor