Machine Learning Lab
Exercise 4 (Week 5):
Pandas

1. Copy file abc.xlsx locally.
2. Use following url for csv (comma separated)file:
   - https://tinyurl.com/titanic-csv
3. Use following url for csv (semicolon separated) file:
   - https://tinyurl.com/yx3b6sq3
4. Use files mentioned in 1 or 2 or 3, for exercises given below.

5. Installing pandas in python
   - Open (Windows) command prompt
   - Type: pip install pandas

6. Try reading excel file from python script
   - data=pd.read_excel(‘Fullpath\abc.xlsx')
   - This will give an error for XLRD

7. Install xlrd if required
   - Type pip install xlrd

8. Now try reading xlsx file from python script this will do

9. Try reading EXCEL file which is in python folder
   - data=pd.read_excel(‘abc.xlsx')

10. Try reading EXCEL file which is in any folder
    - data=pd.read_excel('Fullpath \abc.xlsx')
      - ⊗ This will give an error – Use of character ‘r’ before PATH name
    - data=pd.read_excel(r'Fullpath\abc.xlsx')

11. Reading data from any sheet of given EXCEL file by its name or number
    - data=pd.read_excel(r'D:\Office_PC\abc.xlsx',sheet_name=1)
    - data=pd.read_excel(r'D:\Office_PC\abc.xlsx',sheet_name=’Sheet1’)
    - data=pd.read_excel(r'D:\Office_PC\abc.xlsx',sheet_name=’Sheet2’)
    - data=pd.read_excel(r'D:\Office_PC\abc.xlsx',sheet_name=0)

12. Displaying selected columns
    - print(pd.DataFrame(data,columns=['Eno','Marks1']))
    - make note of name of the functioin DataFrame (D and F is in capital letters)
    - Other way :
      print(data[['Name','Survived']].head())

13. displaying columns using dot operator

- print(data.Eno,data.Marks1)
- This will display two different series

14. Importing data set from URLs

```
import pandas as pd
url='https://tinyurl.com/yx3b6sq3'
data=pd.read_csv(url)
print(data)
```

15. Using different separator from file like ; or other

```
data=pd.read_csv(url,sep=';')
```

16. Using column names explicitly

```
data=pd.read_csv(url,sep=';',names=['CIC0','SM1_Dz(Z)','GATS1i','NdsCH'
,'NdssC','MLOGP','quantitative response LC50'])
```

17. Using any column as an index

- data=pd.read_csv(url,sep=';',index_col=0)

18. Getting selected columns' data from CSV file;

- data=pd.read_csv(url,sep=';', ,usecols=[1,3,4])

19. Prefix to add to column numbers when no header is available

- data=pd.read_csv(url,sep=';',header=None,prefix='Column')

20. Skipping selected rows (callable list)

- data=pd.read_csv(url,sep=';',usecols=[1,5,6],skiprows=[1,3,5])
- data=pd.read_csv(url,sep=';',usecols=[1,5,6],skiprows=lambda x: x%2==0)

Use following options with read_excel/read_csv() with following keyword arguments (Excercise 21 to 24):

21. skipfooter: Number of lines at bottom of file to skip

22. nrows : Number of rows of file to read. Useful for reading pieces of large files

23. na_values : scalar, str, list-like, or dict, optional

24. Additional strings to recognize as NA/NaN. If dict passed, specific per-column NA values. By default the following values are interpreted as NaN: '', '#N/A', '#N/A N/A', '#NA', '-1.#IND', '-1.#QNAN', '-NaN', '-nan', '1.#IND', '1.#QNAN', 'N/A', 'NA', 'NULL', 'NaN', 'n/a', 'nan', 'null'.

25. skip_blank_lines