Module – 1

Importing datasets

1. Important Libraries we'll use:

A. Scientific Computing;

- **Pandas**
  – Data structures and tools for effective data manipulation and analysis
  – 2D tables consisting of colimns and row labels, called data frames
- **NumPy**
  – Mostly for arrays, matrices and vectors in I/O form
- **SciPy**
  – Advanced math problems

B. Data Visulaisation

- **Matplotlib**
  – for graphs and charts
- **Seaborn** *(Made on top of MatplotLib)*
  – heatmaps, violin plots and timeseries

C. Algorithmic Learning

- **SciKit Learn**
  – statistical modeling
  – regressions, classification and clustering
- **Stats models**
  – exploration datasets
  – estimation of statistical models
  – performing statistical tests

2. Import/Export Data in Python

- Data acquisition
  Loading & reading data into notebook from various sources
- Important factors to consider:
  – format of files
  – file path
- in csv, each row is a data point
- csv stands for comma separated values
- .read_csv method can read csv files in pandas

```
import pandas as pd
url = "url_name"
df = pd.read_csv(url) #the shortname we give to our dataframe
```

- This method ***assumes*** that our data contains headers, which is not necessarily true
- If no headers are present:

```
pd.read_csv(url, header=None)
```

– Headers are naturally set to integers in this case

- **Printing Datasets**
  – printing the entire datasets for verification can be a time consuming task
  – we have special functions to limit the no. of printed rows

```
>> df.head(n)
# prints 1st n rows
>> df.tail(n)
# bottom rows
```

- **Headers can be added separately**
  – make a list of headers / store them in a list if if you already have them in some other format and write this line:

```
df.columns = list_name
```

- After having worked with our data frames, we might want to read it into another csv file or store as another csv file

```
path = "path_name_for_saving/file_name"
df.to_csv(path)
```

- The same principles of I/O apply to these formats as well:
  – json
  – excel
  – sql