**Data Analysis handbook**

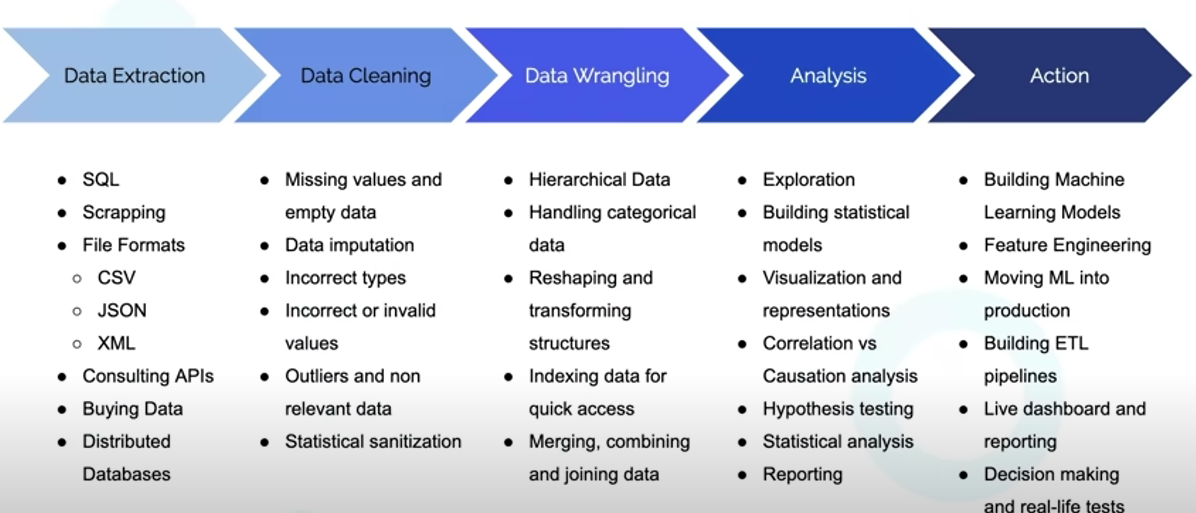
**Data analysis**- It is a collection of procedures or activities to be performed in order to derive useful conclusion from available pool of data gathered, which can be used by various entities for further decision making and commercial purposes.

1. Gathering, inspecting, cleansing and transforming data – this procedure is the most tedious out of all the procedures involving collection of data followed by inspecting what kind of data is gathered (unstructured, semi structured, structured) followed by cleansing the data by making it suitable for further usage (handling missing values, incorrect values, duplicity, etc) followed by transformation of the resultant data to a format suitable for further analysis.
2. Modelling data – this procedure involves adapting real life scenarios to the information systems (using transformed data (drawn from real world scenarios) and statistics to analyse and draw patterns) by using inferential statistics (a way to make predictions from a small set of data and generalize the prediction to larger data or group) and figuring out whether a model or pattern arises.
3. Discovering useful information – this procedure involves drawing conclusions, looking for interesting patterns or anomalies. In this procedure the data gets converted to understandable information
4. Informing conclusions and supporting decision making – This procedure involves creating evidences of our conclusion and finding, creating reports, dashboards and aiding other departments with the gathered information. These analysis’ will be further used by various entities such as marketing sales, accounting executives, etc which will require this analysis from different perspectives and for different requirements and make decisions on basis of these analysis.

The above can be summarised in one definition as follows:

*Data analysis is the process of gathering, inspecting, cleaning, transforming and modelling data with the goal of discovering useful information, informing conclusions and supporting decision making.*

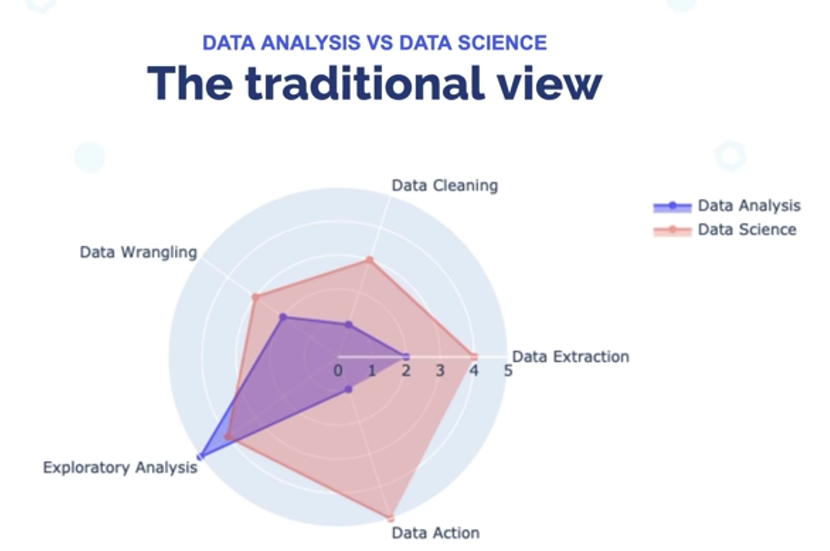
**The processes in data analysis**



**Different tools available for data analysis**

|  |  |
| --- | --- |
| **Auto manged closed tools** | **Programming languages** |
| These involves tools such as Qlik, excel, tableau, etc | These involve languages such as python, R, Julia. |
| These are closed source tools supported and maintained by companies | These are open-source tools |
| These are limited to the functionalities already available in the tools hence not flexible (disadvantage) | These are extremely powerful with a vast pool of open-source libraries and flexibility of creating functionalities other than those available in the libraries (advantage) |
| These are expensive (disadvantage) | These are mostly free or are available at very low prices. |
| These are very easy to learn and use (advantage) | These require understanding of programming basics (disadvantage for non-programmers) |
| A constant visual reference of data is always available and its about point and click. But this doesn’t work well for a very large no. of records | There is no constant visual reference of data. The data, format of data and its properties are known but there is no constant visual reference on the screen. Hence easier to process large records. |
| More overhead to move data analysis process from one computer to another or on cloud | Less overhead to move data analysis processes from one computer to another or on cloud. |

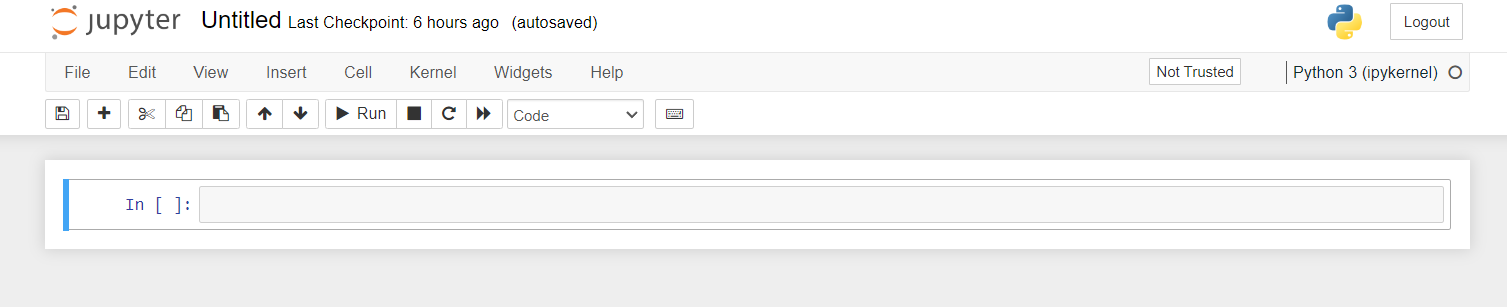
**Difference between data science and data analysis**



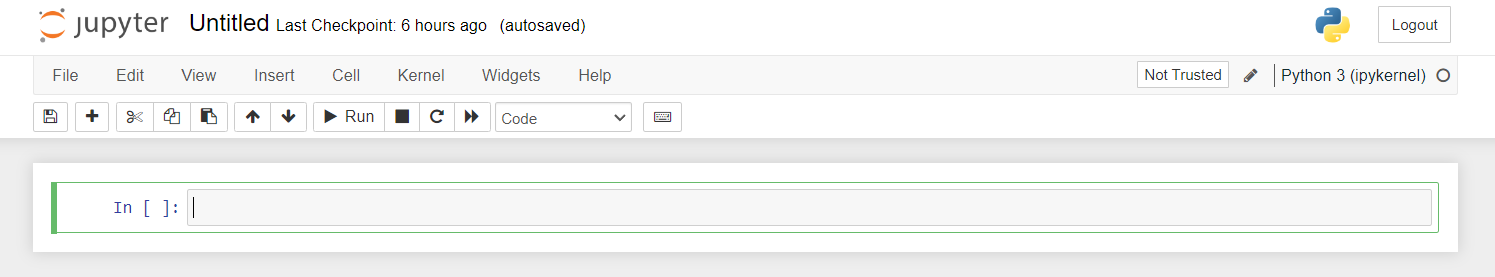
The major difference in data analysis and data science is that data science involves more around mathematics and statistics and requires almost equal understanding of all the processes basic processes however data analysis is more on the analysis side and requires one to be good at forming records of conclusions and predictions.

**Working with Jupyter Notebook**

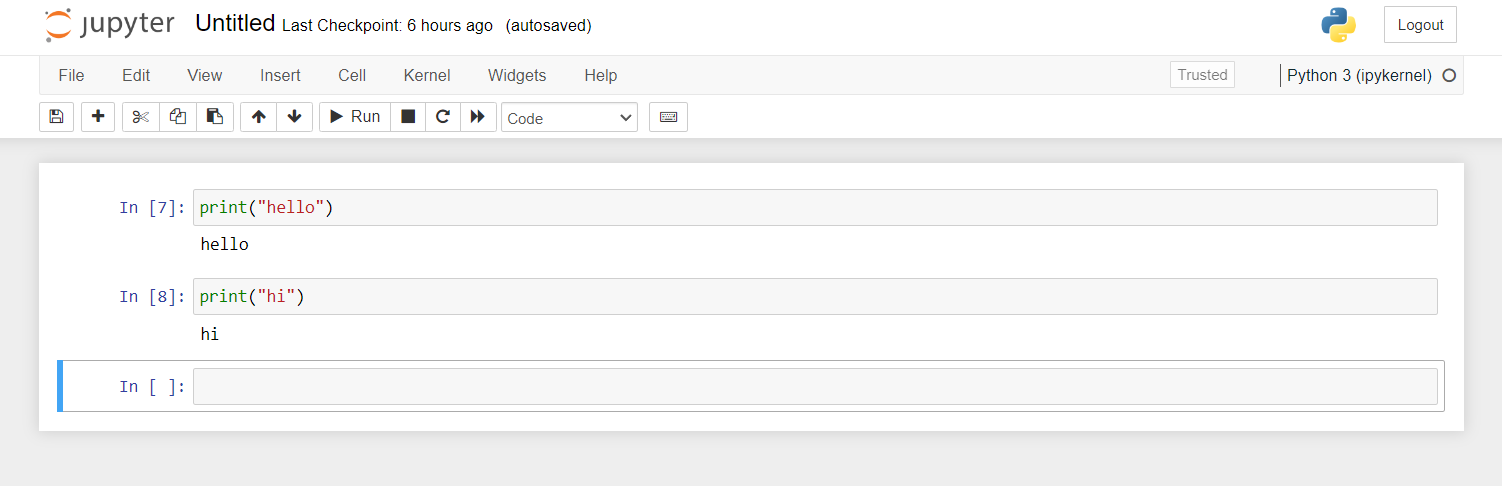
* It is an interactive computational environment that provides a web-based interface, which combines code, output and documentation within a single document.
* Jupyter Notebooks also support the visualization of data and plots, integration with interactive widgets, and the ability to incorporate multimedia elements like images, videos, and audio.
* This the basic Jupyter notebook interface:



* The workspace consists of cells that supports code (Python, R, Julia) , markdown (text formatting language) and raw text.
* The workspace has 2 modes command mode (press esc) and edit mode (press enter or return). The above image shows the command mode depicted by the blue line. The image below shows the edit mode depicted by a green line.



* The command mode allows performing functionalities such as adding cell, deleting cell, etc either by clicking on drop down menu or using keyboard shortcuts.
* The edit mode allows to work within the cells.
* The numbers within the square brackets denotes the order in which the cells were executed. There were in total 8 executions with the first cell last executed on 7th execution and the second last executed on the 8th execution.



* Through Jupyter notebook one creates different instances for different directories, reason being security and access restrictions for other directories. The directories outside root directories cannot by default access the files inside the same instance. To maintain this security, one could create different instances for different directories. But this default functionality of access denial for outside directories can be changed by the following command in terminal:

*jupyter notebook --NotebookApp.allow\_origin='\*' --NotebookApp.allow\_root=True*

The --NotebookApp.allow\_origin='\*' flag allows access from any domain, and the --NotebookApp.allow\_root=True flag enables access to files outside the root directory.

* Keyboard shortcuts:

a – add cell above

b – add cell below

dd – remove current cell

up and down arrows – switch between cells

m – change the edit mode to markdown

p – change the mode to Python

ctrl + enter/return – execute code and remain in same cell

shift + enter/return – execute code and move to next cell

z – undo

x – cut

v – paste

These are common one, others can be looked in the shortcut list on the bar in Jupyter notebook

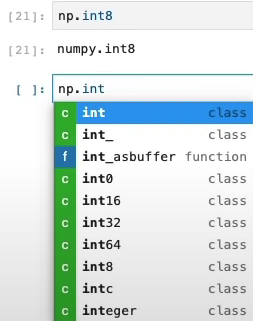
* Jupyter notebook vs Python terminal:

1. Jupyter notebook offers visualizations
2. Jupyter notebook already consists of all the libraries of python so there is no need to explicitly install them
3. Jupyter notebook allows to request data from web APIs
4. Jupyter notebook provides with a library *Bokeh* that allows creating interactive charts and visualizations which are dynamically generated through javascript
5. Jupyter notebook can work well with a wide range of formats such as csv, excel, xml, xls, json files, etc

How to open Jupyter notebook from git directly in google collab. (To be done later)

**Introduction to NumPy**

* One of the most important and old libraries for data processing in python ecosystem
* Its stands for numeric python and is used extensively for numeric computing
* Why NumPy? Because python is slow in processing numbers, NumPy solves that problem for python
* Libraries such as pandas and matplotlib are built on top of NumPy and rely on NumPy for their numeric processing
* In python there is a fixed size of 4 bytes for integer numbers however NumPy gives us the freedom to fix specific bits for our numbers thus saving extra space which is helpful while dealing with a lot of data



* Python’s inbuilt data structures such as list and dictionaries are not high performance in numeric computing. List might not store the list items in contiguous location. NumPy is an array processing library and thus providing contiguous storage or array items. NumPy arrays are high performance
* NumPy utilises the low-level CPU instructions/directives for matrix manipulation thus making the computation very fast. Python does not

Hands on NumPy Jupyter notebook is available in the “data analysis (ipynb files)” directory in d drive.