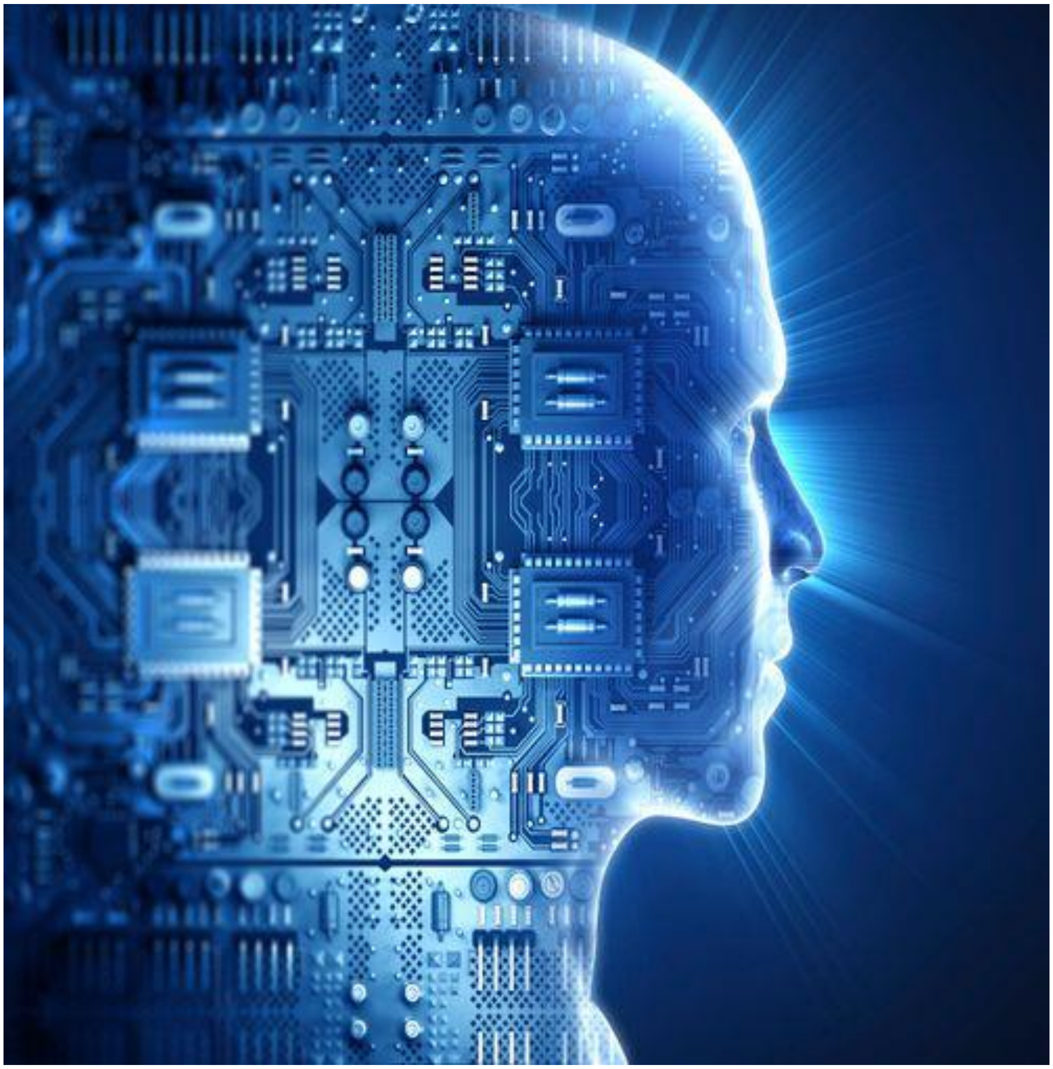


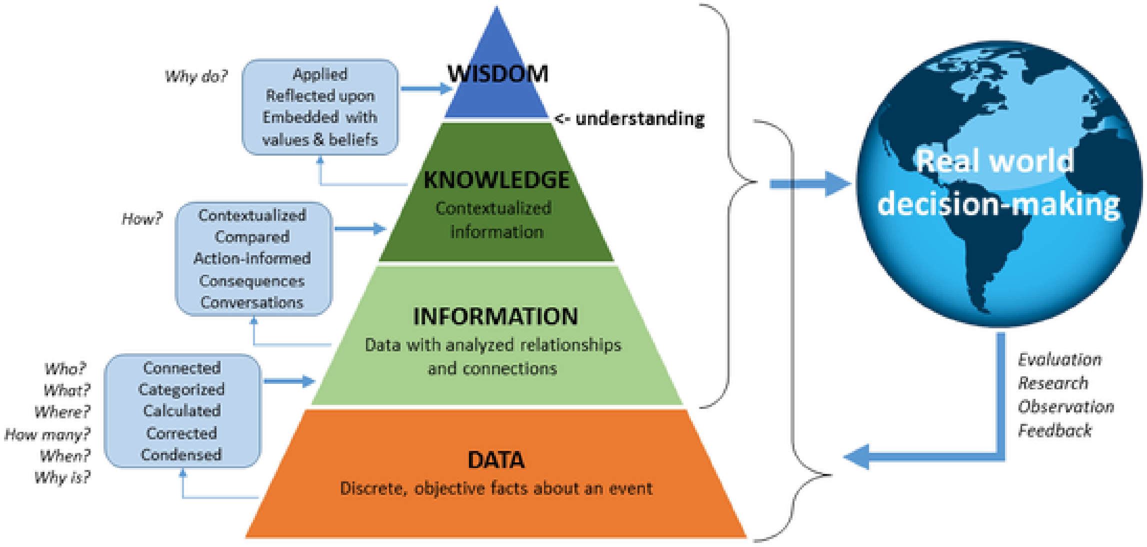
DS – AI - ML - DL

MY LEARNINGS…

Raghunath Dayala | Foundations of DS-AI-ML | 27th Aug,2018

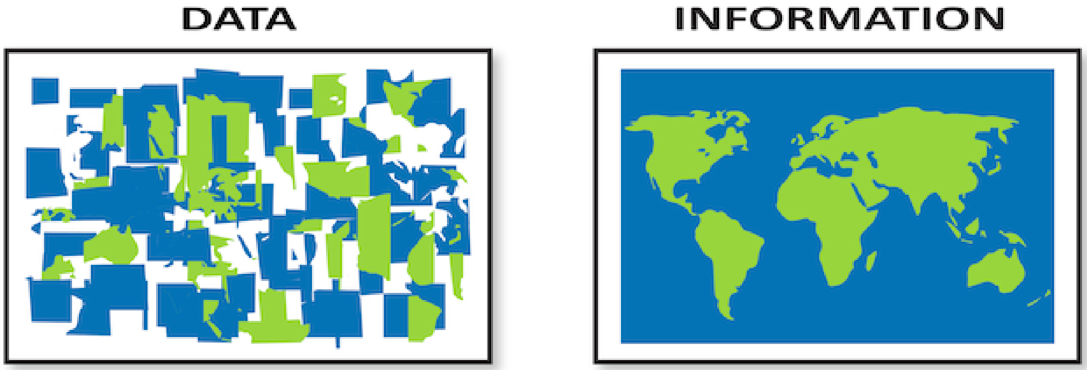
Before we embark on our journey to become a Data Scientist, or ML Engineer or a practitioner of AI, we need to understand what is meant by Data? The very basic question.

What is Data? Data Vs Information?



Data is a collection of raw facts and figures. It can be any character, text, words, numbers, images, audios or videos. If it is not put into any context, it has no meaning to a human.

The data always exists. But it has no value until it becomes *information*, that is until you can consume it in a form that allows you to make a decision.



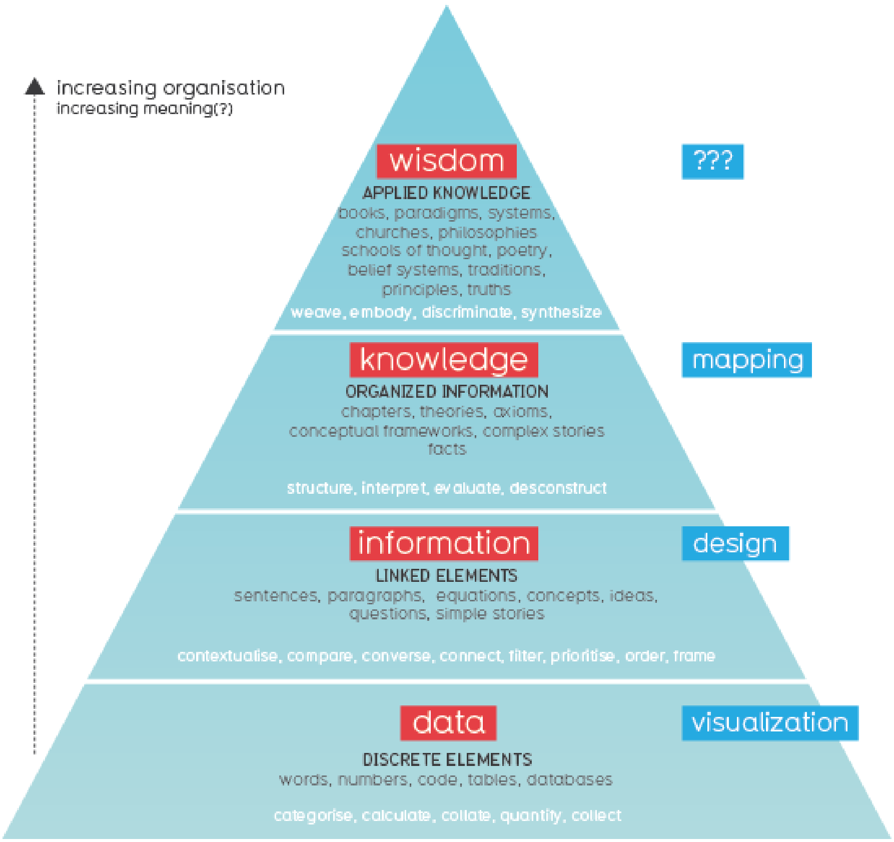
Let’s take a very simple example:

Say, you have a number 500076. This is nothing but Data in numerical form. Does it convey any meaning to you? No, you will just think of it as some random number.

But if I say, this number 500076 represents PINCODE of a city, then that is nothing, but you have information about PINCODE of some city.

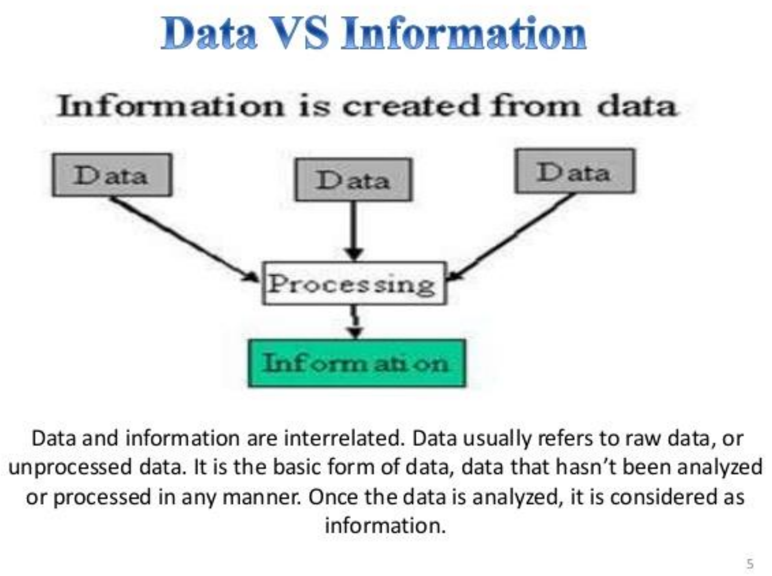
Hierarchy Of Visual Understanding?

Just playing. Something in this?


increasing organisation
increasing meaning(?)
wisdom
APPLIED KNOWLEDGE
books, paradigms, systems, churches, philosophies schools of thought, poetry, belief systems, traditions, principles, truths
weave, embody, discriminate, synthesize
???
knowledge
ORGANIZED INFORMATION
chapters, theories, axioms, conceptual frameworks, complex stories facts
structure, interpret, evaluate, desconstruct
mapping
information
LINKED ELEMENTS
sentences, paragraphs, equations, concepts, ideas, questions, simple stories
contextualise, compare, converse, connect, filter, prioritise, order, frame
design
data
DISCRETE ELEMENTS
words, numbers, code, tables, databases
categorise, calculate, collate, quantify, collect
visualization

David McCandless // v 0.1 // work in progress

InformationIsBeautiful.net

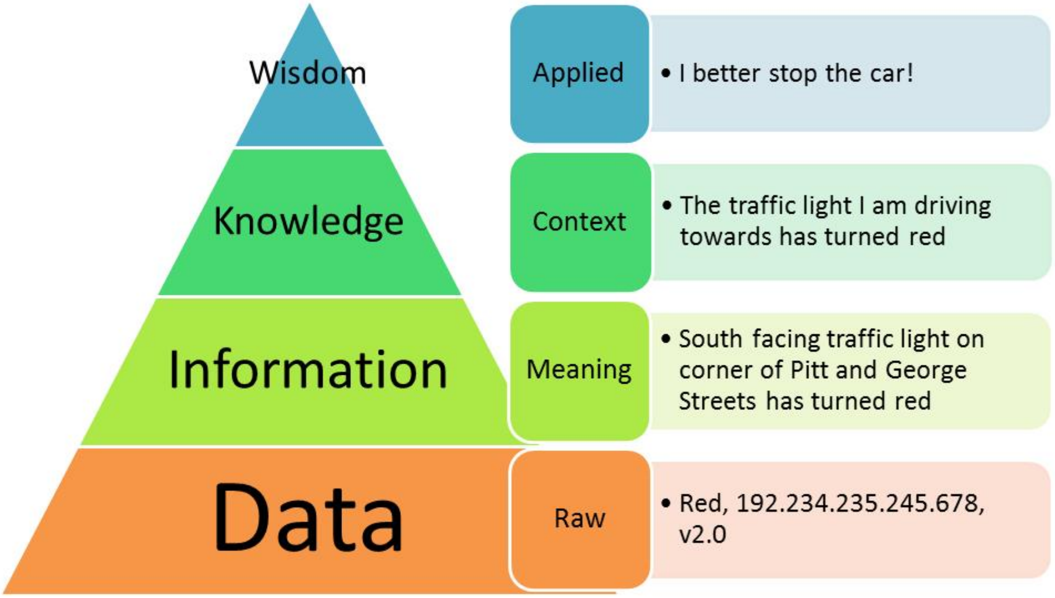


Data VS Information

Information is created from data

Data and information are interrelated. Data usually refers to raw data, or unprocessed data. It is the basic form of data, data that hasn't been analyzed or processed in any manner. Once the data is analyzed, it is considered as information.

Another Example:



Wisdom	Applied	• I better stop the car!
Knowledge	Context	• The traffic light I am driving towards has turned red
Information	Meaning	• South facing traffic light on corner of Pitt and George Streets has turned red
Data	Raw	• Red, 192.234.235.245.678, v2.0

© 2011 Angus McDonald

Data – many Buzz words?? Why are there so many?

Whether you take it or not, there is no denying that, Data is at the foundation of any successful company. Looking deeper into the data is what will make the companies tower above the competition.

Centered around data, there are many buzzwords. Data, Data team, Big Data team, Business Intelligence, Data science, Data analytics, Business analytics, etc.

Little brief of these terms:

- **Data team/Big data team** – this team is the one that gets in first. They will do a significant amount of work on the available data first. (Data preprocessing)

- **Business Intelligence team** – Based on the data provided by Data team, this team will provide business insights dashboard.

- **Data science team** – After the dashboards are ready, this team will use some business analytics or data analytics tools to develop models that could predict future outcomes.

One source of confusion of these buzz words is the constant evolution of Data Science industry.

Long ago there used to be a **Statistician** – responsible for:

- *Gathering and cleaning data sets*
- *Applying statistical models*

(With the growth of Data, radical improvement of Technology)

Data Mining specialist – responsible for:

- *Extracting patterns from Data*

(With the advent of new Mathematical and Statistical models)

Predictive Analytics Specialist – responsible for:

- *Performing more accurate forecasts*

In today's world, we refer to the person who performs all this as a **Data Scientist**.

Also, we often heard of the terms, Data Analysis/Business Analysis, Data Analytics, Business Analytics like this. Hey hold on. Why are we using two terms Analysis or Analytics? Aren't they same?

Analysis Vs Analytics:

Say, you have a huge dataset containing data of various types (ID, name, etc). Instead of studying the entire data set, you separated the dataset into small chunks and study them individually and examine how they relate to other parts. This is called Analysis.

You perform **Analysis** on things that have already happened in the **Past**.

Example:

You watched a movie and then you are explaining why the movie ended that way. What events happened in the movie and how they relate to each other.

All this means, we do ***analysis to explain how and or why something happened*.**

Analytics generally refers to the **future**, instead of explaining past events.

Analytics is essentially, the *application of logical and computational reasoning* to the observations you did in an Analysis. In effect, you are looking for patterns to explore potential future events.

1. ***Qualitative Analytics*** – Intuition (Experience) + Analysis

   Here you use, your intuition and experience in combination with the Analysis to plan your next business move.

2. ***Quantitative Analytics*** – Formulas + Algorithms

   This is like applying formulas and algorithms to the numbers that you have gathered from Analysis.

Example:

Say you are an owner of an online clothing store. Based on your experience, you have a good understanding of the needs of your customers (Intuition /Experience). You've also performed a very detailed analysis from different magazine, articles and you are sure which fashion trends are in demand.

Pause a bit. Think of what is it that you are doing here. You are using your intuition and analysis to predict on the new styles of clothing to start selling. This is '**Qualitative Analytics**'.

But at this moment, you do not know when to introduce the new collection. Relying on your past sales data, user experience data, you could predict in which month it would be best to introduce your new collection. This is '**Quantitative Analytics**'.

What it is like being a Data scientist/ML Engineer? How can we add value while working with Data??

There are 3 main things that we always look up to when working as a Data scientist:

- Data
- Algorithm
- Insight

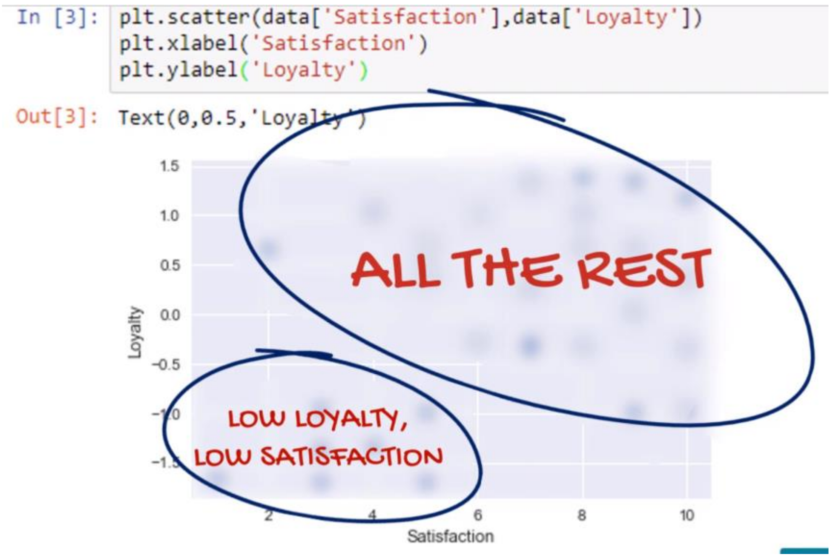
Example:

Say, you have data collected from the customers of a shop. The data contains around 30 observations in total and each observation represents a customer who shared their *customer satisfaction* rating and *brand loyalty* w.r.t the shop.

Let's suppose the owners of a shop hired you as a Data scientist to analyze customer behavior. As a consultant, you have to analyze the data and have to add value to the owners of the shop.

One thing you can do is to, *divide the customer base of the shop into groups of individuals* with similar traits. This will greatly reduce the complexity and then the owners of the shop can think of ways to serve these customer groups better and win the business in the long run. (Q: What is the best way to divide the customer base in to groups? **Clustering** technique)

Without applying any Machine Learning algorithms, the simplest way to make sense of the data is to Visualize the Data.



```
In [3]:  plt.scatter(data['Satisfaction'],data['Loyalty'])
         plt.xlabel('Satisfaction')
         plt.ylabel('Loyalty')

Out[3]:  Text(0,0.5,'Loyalty')
```

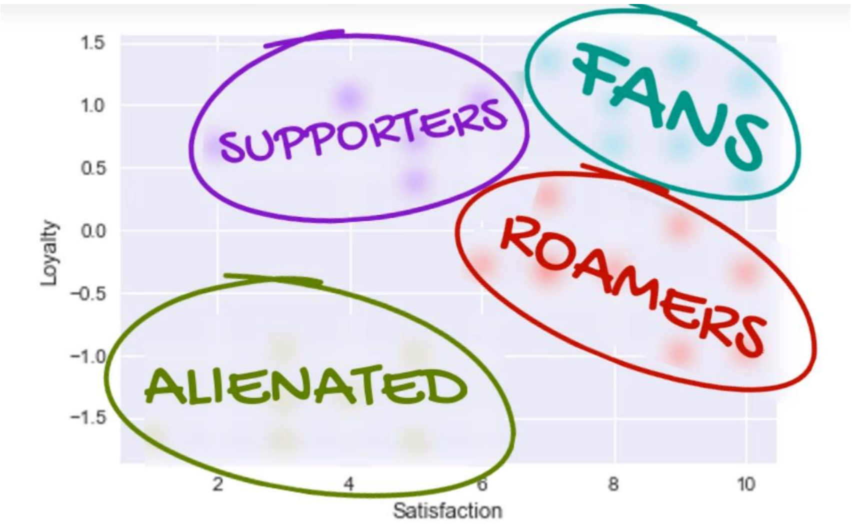
Just looking at the above picture, we can say that there are 2 groups of customers. One group is with Low Loyalty and Low Satisfaction and the other group, the rest of the customers. So far, we have not applied any Machine learning algorithms or anything else. We just visualized the data that we have, and we could divide the customer base into 2 groups.

Now let's apply ML Algorithm. (Don't worry if you don't understand what this code means. It makes sense after a while. We will discuss these code snippets later)

```
X = preprocessing.scale(data)
kmeans = KMeans(4)
kmeans.fit(X)
```

With this, our customer base is now segmented into 4 groups. In fact, we can name those groups.

- Low Satisfaction, Low Loyalty - Alienated
- Low Satisfaction, High Loyalty – Supporters → Dissatisfied Customers
- High Satisfaction, Low Loyalty – Roamers → Disloyal Customers
- High Satisfaction, High Loyalty – Fans



On our **Data**, we applied an **Algorithm** and got an **Insight**. Our customer base is now segmented into 4 groups.

Data Science is about Story telling. You should be able to make sense of numbers.

In above picture, you have 4 groups, but only one of them is Fans. Our data analysis now indicates the problem. Now the question arises, how can we make other groups to become Fans?? How do we solve??

We can now think of better ways to serve these customer groups so that they become fans.

- Supporters (Dissatisfied Customers) → Fans:

Dig deeper to figure out the drivers for Dissatisfaction. May be long queues, high prices or unfriendly staff. Once we know this, we can think of some actionable steps improve these customers shopping experience, so they become satisfied.

- Roamers (Disloyal customers) → Fans:

For these customers, you can think of ways to introduce loyalty cards, or special discounts, vouchers, etc.

Data Science is about working with data, make sense of numbers and add value to an organization in making better and informed decisions.

Data Pre-processing – Importing the Data set:

Your data can be stored in many different files, like text files, CSV files, XLSX files, etc. How can we import data from these different sources? Also, sometimes, you may need to load data from URLs as well.

Check the below link for better understanding:

https://www.analyticsvidhya.com/blog/2017/03/read-commonly-used-formats-using-python/

Loading CSV Files: (https://docs.python.org/2/library/csv.html)

The most common format for machine learning data is CSV files. There are a number of ways to load a CSV file in Python.

There are a number of considerations when loading your machine learning data from CSV files.

- **CSV File Header** - Does your data have a file header?
  - If so this can help in automatically **assigning names to each column** of data. If not, you may need to name your attributes manually.
  - Either way, you should explicitly specify whether or not your CSV file had a file header when loading your data.
- **Comments** - Comments in a CSV file are indicated by a hash (“#”) at the start of a line.
  - If you have comments in your file, depending on the method used to load your data, you may need to indicate whether or not to expect comments and the character to expect to signify a comment line.
- **Delimiter** - The standard delimiter that separates values in fields is the comma (“,”) character.
  - Your file could use a different delimiter like tab (“\t”) in which case you must specify it explicitly.
- **Quotes** - Sometimes field values can have spaces. In these CSV files the values are often quoted.
  - The default quote character is the double quotation marks “\””. Other characters can be used, and you must specify the quote character used in your file.

For example, you can download the Pima Indians dataset into your local directory (update: download from here). All fields are numeric and there is no header line.

Load CSV with Python Standard Library:

Code :

```
# Load CSV (using python)
import csv
import numpy
filename = 'pima-indians-diabetes.data.csv'
raw_data = open(filename, 'rt')
reader = csv.reader(raw_data, delimiter=',', quoting=csv.QUOTE_NONE)
x = list(reader)
data = numpy.array(x).astype('float') # convert to Numpy array
print(data.shape)
```

The example loads an object that can iterate over each row of the data and can easily be converted into a NumPy array.

Load CSV with NumPy library:

Code :

```
# Load CSV using NumPy
import numpy as np
filename = 'pima-indians-diabetes.data.csv'
raw_data = open(filename, 'rt')
data = np.loadtxt(raw_data, delimiter=",")
print(data.shape)
```

You can load your CSV data using NumPy and the ***numpy.loadtxt()*** function.

We can also load the same dataset directly from a URL as follows:

```
# Load CSV from URL using NumPy
import numpy as np
from urllib.request import urlopen
url = 'https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-
indians-diabetes.data.csv'
raw_data = urlopen(url)
dataset = np.loadtxt(raw_data, delimiter=",")
print(dataset.shape)
```

For more information on the numpy.loadtxt() function see the API documentation.

Load CSV with Pandas:

You can load your CSV data using Pandas and the *pandas.read_csv()* function.

This function is very flexible and is perhaps my **recommended approach** for loading your machine learning data. The function returns a pandas.DataFrame that you can immediately start summarizing and plotting.

```
# Load CSV using Pandas, specifying column names
import pandas
filename = 'pima-indians-diabetes.data.csv'
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
data = pandas.read_csv(filename, names=names)
print(data.shape)
```

We can also load CSV data directly from a URL.

```
# Load CSV using Pandas from URL
import pandas
url = "https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-indians-diabetes.data.csv"
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
data = pandas.read_csv(url, names=names)
print(data.shape)
```

To learn more about the pandas.read_csv() function you can refer to the API documentation.

For more examples, check this link:
https://chrisalbon.com/python/data_wrangling/pandas_dataframe_importing_csv/

Load a csv with no headers:

```
df = pd.read_csv('example.csv', header=None)
```

Load a csv while specifying column names:

```
df = pd.read_csv('example.csv', names=['ID', 'First Name', 'Last Name', 'Age', 'Pre-Test Score', 'Post-Test Score'])
```

Load a csv with setting the index column:

```
df = pd.read_csv('example.csv', index_col='ID', names=['ID', 'First Name', 'Last Name', 'Age', 'Pre-Test Score', 'Post-Test Score'])
```

Loading Images:

Matplotlib:

https://matplotlib.org/

https://matplotlib.org/tutorials/index.html

Matplotlib is a Python plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter notebook, web application servers, and four graphical user interface toolkits.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython.

Simple installation:

```
pip install matplotlib
```

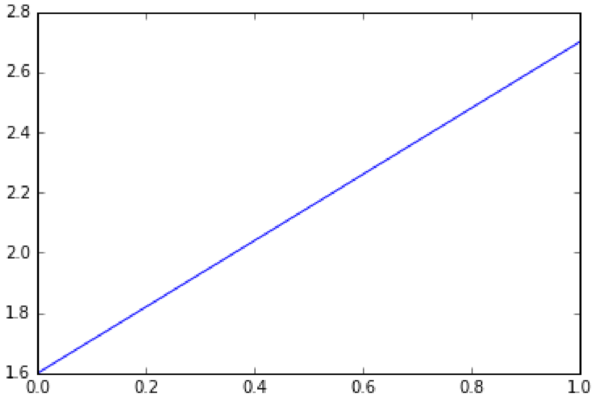
Import matplotlib's pyplot module and display all visuals inline:

```
import matplotlib.pyplot as plt
%matplotlib inline
```

Create a simple plot:

```
plt.plot([1.6, 2.7])
```

```
[<matplotlib.lines.Line2D at 0x10c4e7978>]
```



Example:

```
plt.plot([1,2,3],[5,7,4])
plt.show()
```

We invoke the **.plot()** method of pyplot to plot some coordinates. This .plot takes many parameters, but the first two here are ‘x’ and ‘y’ coordinates, which we’ve placed lists into. This means, we have 3 coordinates according to these lists: (1,5) (2,7) and (3,4).

The plt.plot will “draw” this plot in the background, but we need to bring it to the screen when we’re ready, after graphing everything we intend to.

plt.show() will make the graph pop up and you should be able to see.

Exploratory Data Analysis?

Exploratory Data Analysis (EDA) is the **first step** in your data **analysis** process. Here, you make sense of the data you have and then figure out what questions you want to ask and how to frame them, as well as how best to manipulate your available data sources to get the answers you need.

http://mbcoder.com/exploratory-data-analysis-with-python/

Kaggle is a platform for predictive modeling and analytics competitions in which companies and researchers post data and statisticians and data miners compete to produce the best models for predicting and describing the data.

Heading 1

To replace the placeholder text on this page, you can just select it all and then start typing. But don't do that just yet!

First check out a few tips to help you quickly format your report. You might be amazed at how easy it is.

- Need a heading? On the Home tab, in the Styles gallery, just click the heading style you want.
- Notice other styles in that gallery as well, such as for a quote, a numbered list, or a bulleted list like this one.
- For best results when selecting text to copy or edit, don't include space to the left or right of the characters in your selection.

HEADING 2

You might like the photo on the cover page as much as we do, but if it's not ideal for your report, it's easy to replace it with your own.

Just delete the placeholder picture. Then, on the Insert tab, click Picture to select one from your files.