Today's agenda

- Sample and Population
- Merits and Demerits of sampling
- Types of sampling
- Random sampling
  - Implementing the same using pandas
- Predictive analytics
- Importance of sampling in PA

Sample Population & Sampling

Population

- A set of similar items or events which is of interest for some question or experiment.
- We denote the population as `N`.

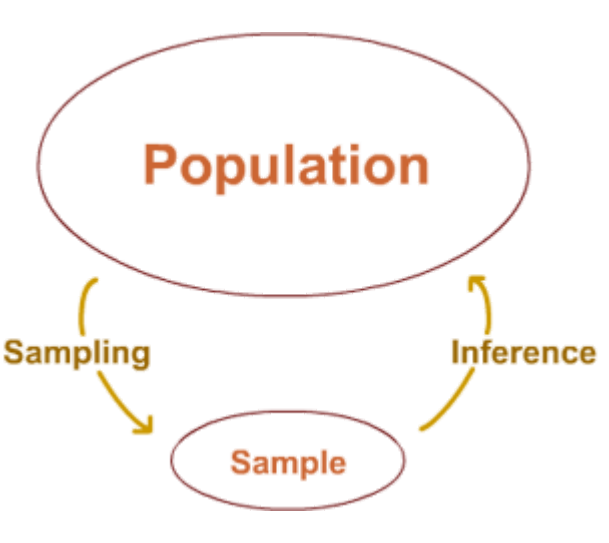
Sample

- A subset of the population (a statistical sample) that is chosen to represent the population.
- We denote the sample as `n`.

Sampling

(method)

- A selection of subset of individuals from statistical population to estimate the characteristics for the whole population.
- It is one such technique that is applied by everyone in our day to day activities.



Credits - Image from Internet

Note

- By taking sample, statisticians tend to infer or conclude the characteristics/estimates to the whole population.

Example

- Imagine you have a piece of land and you want to know if the land is fertile enough to grow plants.

- **Scenario 1**
  - Interpret the land's fertility by testing the whole land.
- **Scenario 2**
  - Interpret the land's fertility by just testing a sample (soil) in a container or jar.

Credits - Image from Internet

Merits & Demerits

Merits

- Less cost effective
- Time saving
- Higher accuracy

Demerits

- Chances of biasness
- Need of subject specific knowledge

Types of Sampling

1. **Probability Sampling**
   - `Simple Random Sampling`
     - It is a randomly selected subset where each member of the population has an exactly equal chance of being selected.
     - From the random sample that is selected, researcher tends to make statistical inferences on the whole population.
   - Systematic Sampling
   - Cluster Sampling
   - Stratified Sampling
2. **Non-Probability Sampling**
   - Convenience Sampling
   - Judgmental Sampling
   - Snowball Sampling
   - Quota Sampling

What should be the size of the sample?

- Refer to → https://www.tools4dev.org/resources/how-to-choose-a-sample-size/

In [1]:
```
import pandas as pd
import numpy as np
```

Population data

Get random integers in the range of `low` and `high`

- size → (how_many_rows, how_many_columns) - (10000, 3)

Make random data using pandas

In [2]:
```
# help(np.random.randint)
```

In [3]:
```
# popn_data (population)
popn_data = np.random.randint(low=10, high=1000, size=(10000, 3))
```

In [4]:
```
# display popn_data
popn_data
```

Out[4]:
```
array([[124, 287, 466],
       [651, 231, 681],
       [557, 667, 278],
       ...,
       [ 91, 593, 423],
       [339, 266, 251],
       [539, 446, 403]])
```

Create a dataframe with columns and data generated

In [5]:
```
# df
df = pd.DataFrame(data=popn_data, columns=['col_1', 'col_2', 'col_3'])
```

In [6]:
```
# head()
df.head()
```

Out[6]:

	col_1	col_2	col_3
0	124	287	466
1	651	231	681
2	557	667	278
3	577	600	741
4	54	150	274

Population data (df) size is 10000

- N = 10000

In [7]:
```
df.tail()
```

Out[7]:

	col_1	col_2	col_3
9995	917	482	431
9996	618	996	650
9997	91	593	423
9998	339	266	251
9999	539	446	403

In [8]:
```
# shape
df.shape
```

Out[8]:
```
(10000, 3)
```

Random selection without `pandas`

In [9]:
```
rand_locs = np.random.randint(low=0, high=df.shape[0], size=(100, ))
```

In [10]:
```
rand_locs
```

Out[10]:
```
array([3796, 1510, 9587, 4037, 4906, 4978, 4465, 1039, 7940, 6726, 8202,
       1175, 1218, 7237, 4770, 6296, 3533, 6853, 7639, 3624,  723, 9667,
       8391, 3329, 1771, 1138, 1109, 9713, 5788, 2873,  446, 6019,  609,
       3409, 4910, 7734, 4116, 8840, 9643, 7053,   37, 9220, 7623, 4844,
       4720, 2216, 6977, 2690, 9081,  311, 2494, 7638, 9736, 1416, 9547,
       9022, 1882, 7016, 4303, 4788, 8426, 4871, 1686, 1206, 4667, 6404,
       2650, 8622, 6746, 8158, 2296,  355, 9579, 5405, 1738, 7467, 6030,
       5424, 3761, 2669,  180, 1416, 1677, 7559,  486, 6055, 1438, 6985,
       1618, 5097, 5050,  789,  651, 4406, 9399, 5783,  778, 7604, 4732,
       7323])
```

In [11]:
```
rand_sample_df = df.loc[rand_locs]
```

In [12]:
```
rand_sample_df.head()
```

Out[12]:

	col_1	col_2	col_3
3796	207	171	422
1510	436	473	437
9587	71	559	190
4037	936	476	144
4906	467	20	468

Simple random sample

- Select a sample dataframe from population (df) of size 100
- n = 100

In [13]:
```
# dir(df)
```

In [14]:
```
# ??df.sample
# help(df.sample)
```

In [15]:
```
# rand_sample_df
# dir(df)
rand_sample_df = df.sample(n=100, random_state=42)
```

In [16]:
```
# shape
rand_sample_df.shape
```

Out[16]:
```
(100, 3)
```

In [17]:
```
# head
rand_sample_df.head()
```

Out[17]:

	col_1	col_2	col_3
6252	675	955	81
4684	158	513	645
1731	777	816	474
4742	390	89	597
4521	714	39	426

A descent way of sampling can be achieved by `frac`

In [18]:
```
# frac
random_sample_df_f = df.sample(frac=0.1, random_state=20)
```

In [19]:
```
# head
random_sample_df_f.head()
```

Out[19]:

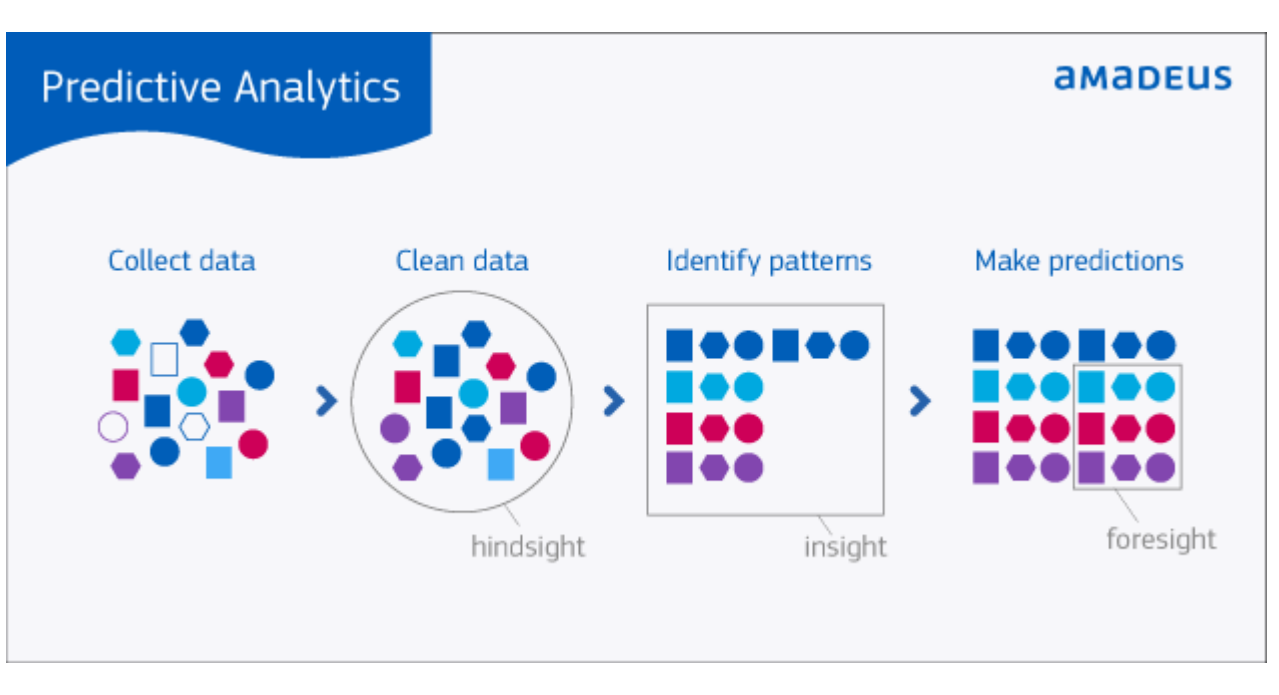
	col_1	col_2	col_3
9957	733	89	948
1687	304	223	674
2116	552	156	216
231	488	596	880
2780	654	230	592

In [20]:
```
# shape
random_sample_df_f.shape
```

Out[20]:
```
(1000, 3)
```

Predictive Analytics

Predictive analytics encompasses a variety of statistical techniques from `data mining`, `predictive modelling`, and `machine learning`, that analyze current and historical facts to make predictions about future or otherwise unknown events.



Credits - Image from http://bigdata-madesimple.com/

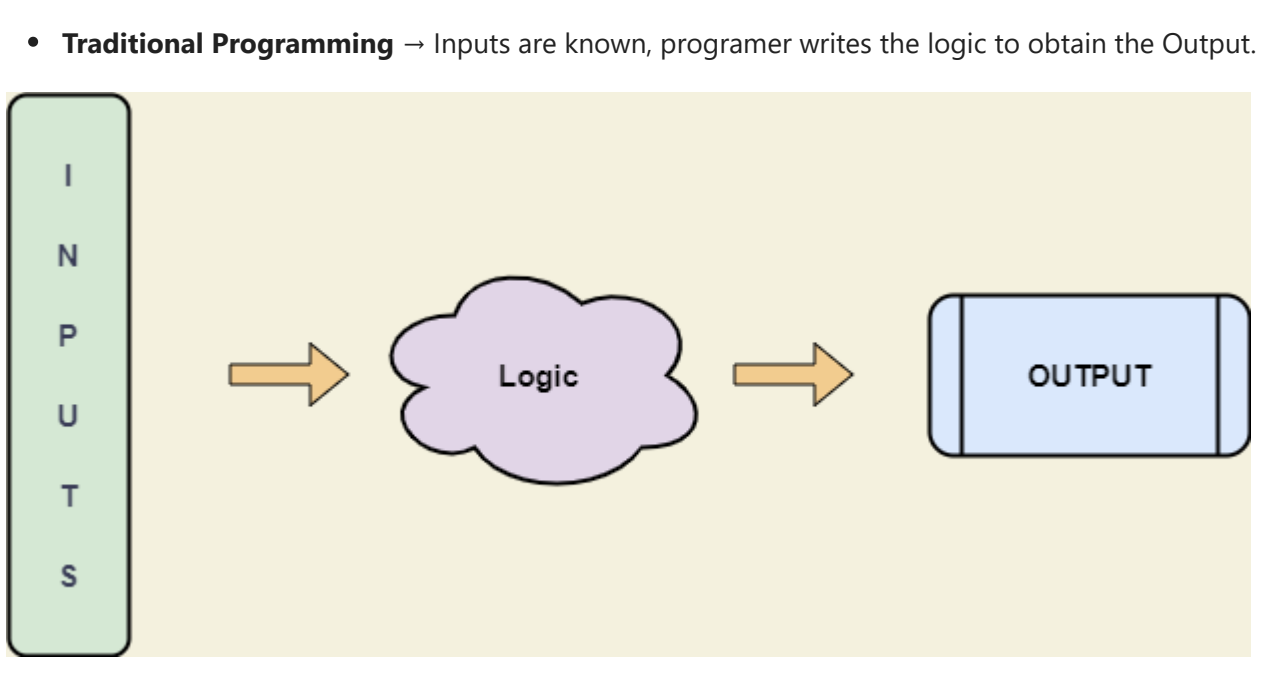
Machine Learning

- ML is a technique followed to make a computer learn from the previous experience in order to make an assumption for the future outcome.
- It can learn and adapt to the new data without any human intervention.
- It needs prior training so that it can be tested to the new data.

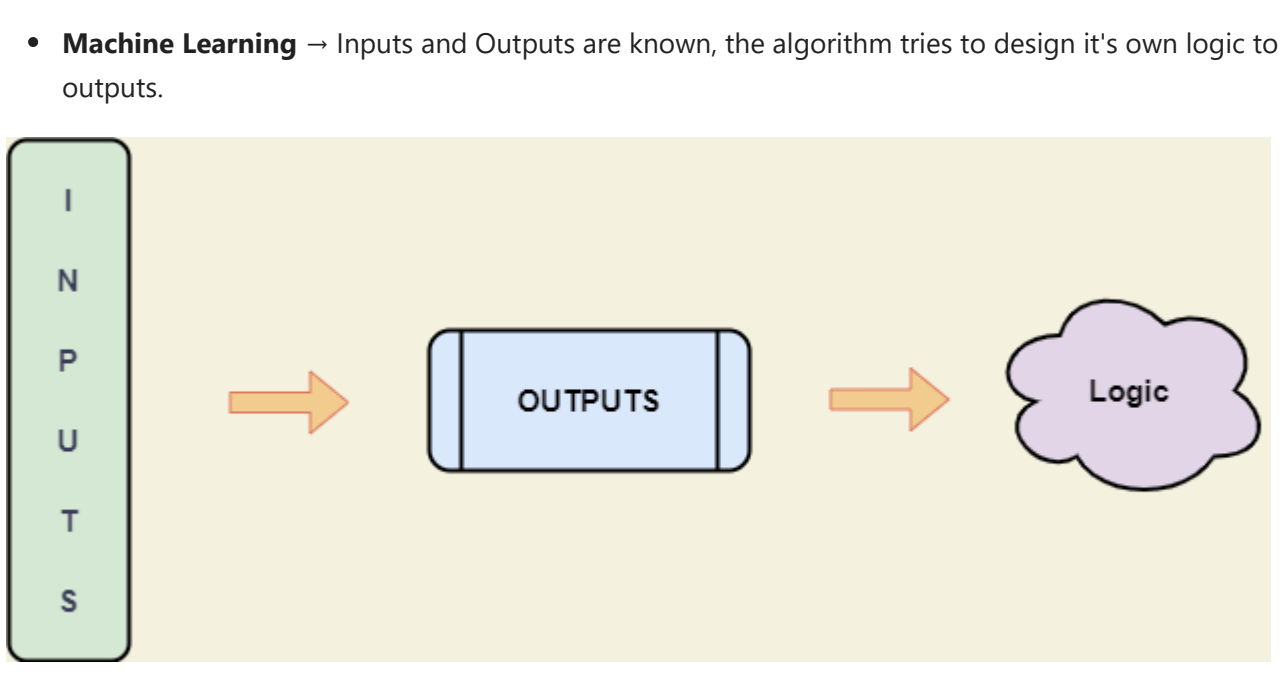
What is this???

ML and Traditional Programming

- **Traditional Programming** → Inputs are known, programer writes the logic to obtain the Output.



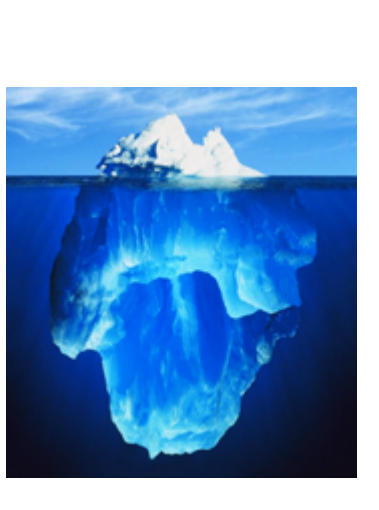
- **Machine Learning** → Inputs and Outputs are known, the algorithm tries to design it's own logic to map the inputs with the outputs.



Images by Author

ML and Mathematics

- ML is just the tip of the iceberg.
- Math and Python code (algorithms) holding the iceberg is what we should be understanding.



Credits - Image from Internet

Examples

- Email spam detector
- Auto-completion mode in the email
- Google photoes classification
- Weather forecasting - Time series prediction
- ...

Types of ML

- **Supervised Learning**
  - The computer is presented with both example inputs and their respective outputs. The algorithm learns a general rule to map the inputs with the outputs.
- **Unsupervised Learning**
  - No outputs are given to the learning algorithm, instead the algorithm alone has to figure out the structure in the inputs and find the hidden patterns to get the final end.
- **Reinforcement Learning**
  - Works based on the reward system and the ultimate goal is to maximize the reward score.

How much data do you really need for building a predictive model?

Often times, we have been told that - to build a machine learning predictive model, we need to have large amounts of data. Well that depends ultimately.

- Effective sampling is about maximizing the about (information) of the whole population from the sampling unit.
- A small random probability sample, as long as it is truly random and not biased in any way, can have very high predictive power.
- With less also, you can achieve more.