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 - Predictive Analytics

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

In [ ]:

Merits & Demerits

Merits

- Less cost effective
- Time saving
- Higher accuracy

Demerits

- Chances of biasness
- Need of subject specific knowledge

In [ ]:

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

In [ ]:

In [11]:
```
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) - (1000, 3)

Make random data using pandas

In [21]:
```
# rand_data (population)
rand_data = np.random.randint(low=5, high=110, size=(1000, 3))
```

In [31]:
```
# display rand_data
rand_data
```

Out[31]:
```
array([[85, 19, 29],
       [19, 36, 86],
       [79, 79, 97],
       ...,
       [23, 14, 28],
       [56, 74, 89],
       [41, 10, 33]])
```

Create a dataframe with columns and data generated

In [43]:
```
# df
df = pd.DataFrame(data=rand_data, columns=['col_1', 'col_2', 'col_3'])
```

In [51]:
```
# head()
df.head()
```

Out[51]:

	col_1	col_2	col_3
0	85	19	29
1	19	36	86
2	65	79	97
3	46	46	45
4	45	70	70

Population data (df) size is 1000

- N = 1000

In [61]:
```
# shape
df.shape
```

Out[61]:
```
(1000, 3)
```

Simple random sample

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

In [81]:
```
# rand_sample_df
# dir(df)

rand_sample_df = df.sample(n=100, random_state=2)
```

In [91]:
```
# shape
rand_sample_df.shape
```

Out[91]:
```
(100, 3)
```

In [101]:
```
# head
rand_sample_df.head()
```

Out[101]:

	col_1	col_2	col_3
37	88	45	14
729	50	16	24
846	7	10	49
296	21	97	40
934	97	94	18

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

In [123]:
```
# frac
# help(df.sample)

rand_sample_df = df.sample(frac=0.5)
```

In [131]:
```
# shape
rand_sample_df.shape
```

Out[133]:
```
(500, 3)
```

In [143]:
```
# head
rand_sample_df.head()
```

Out[143]:

	col_1	col_2	col_3
157	14	44	74
384	33	84	27
493	31	27	37
442	10	31	52
729	50	16	24

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

In [ ]:

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://srgdata.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.

In [ ]:

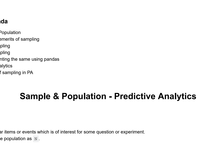
What is this???

In [ ]:

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

In [ ]:

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
- ...

In [ ]:

Types of ML

- **Supervised Learning**
  - The machine is presented with both example inputs and their respective outputs. The algorithm learns a general rule to map the inputs to 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.

In [ ]:

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.

More information -> https://www.sv-europe.com/blog/predictive-analytics-much-data-really-need/