SAP | edunet foundation | code unnati

Unit 1

# Machine Learning: Introduction

## Disclaimer

The content is curated from online/offline resources and used for educational purpose only

# Machine Learning: Introduction





## Learning Objectives

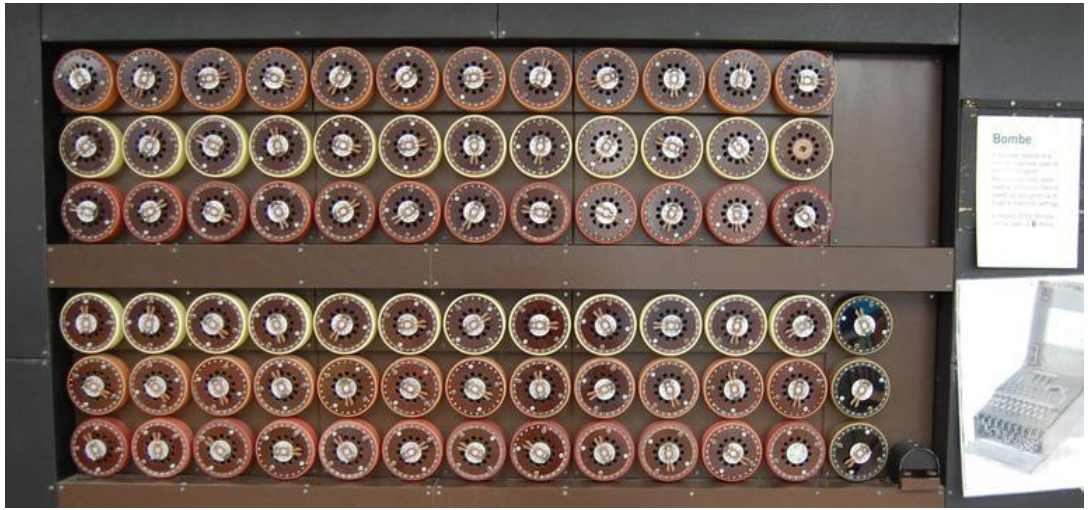
- About Machine Learning
- Applications
- Types of Platforms
- Categories of Machine Learning
- GUI & Non-GUI Approach
- "Hello world to ML"
- Git & GitHub

## About Machine Learning

Concept emerged during WW-II

Primarily knows as Turing Machine

- Intent to learn encrypted message.
- Accepted as field of science in 1950's.



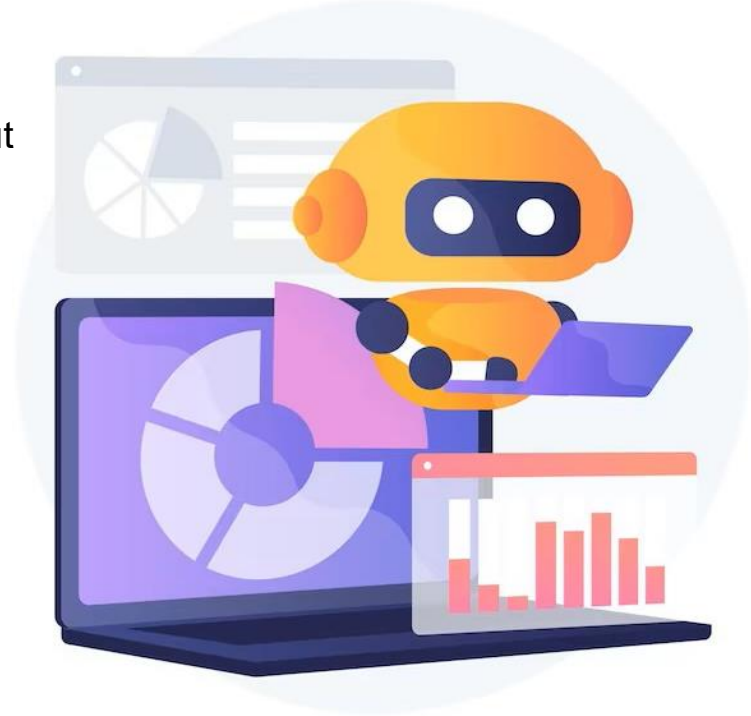

Reference link

## Machine Learning: Definition

Early definition of Machine Learning

“Field of study that gives computers the ability to learn without being explicitly programmed”.  Arthur Samuel (1959)

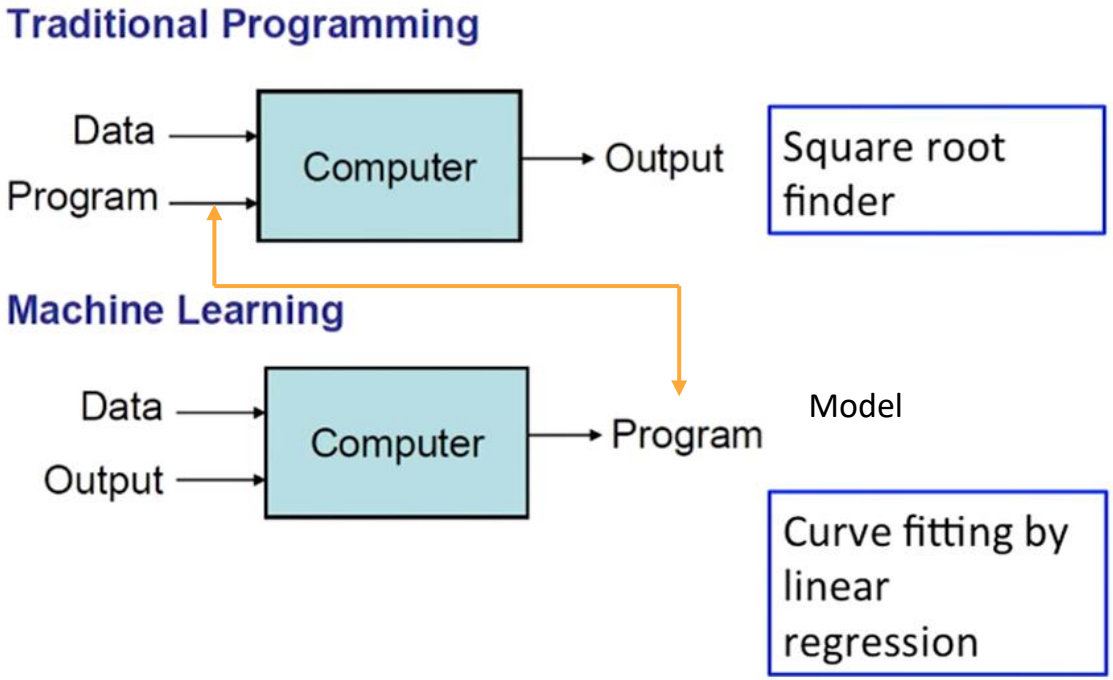
- What do you mean by Explicitly Programmed ?
- So, machine learning algorithms, inspired by the human learning process, iteratively learn from data, and allow computers to find hidden insights.
- These models help us in a variety of tasks, such as object recognition, summarization, recommendation, and so on.



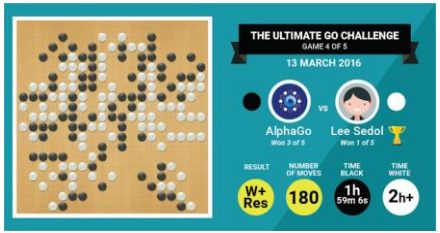
Click here

Reference link

## Differs to Casual Programming



## Machine Learning Applications

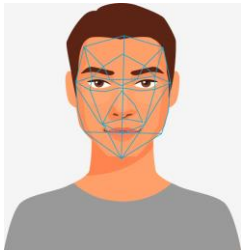



AlphaGo (deepmind.com)

NETFLIX

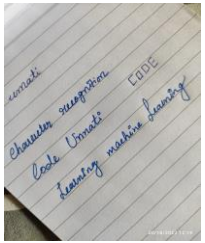
Recommendation System

Drug Discovery

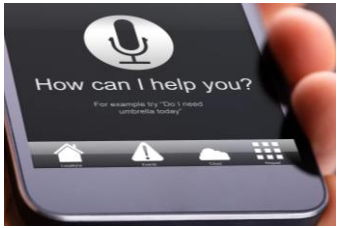


Facial Recognition

Assisted Driving



Character Recognition



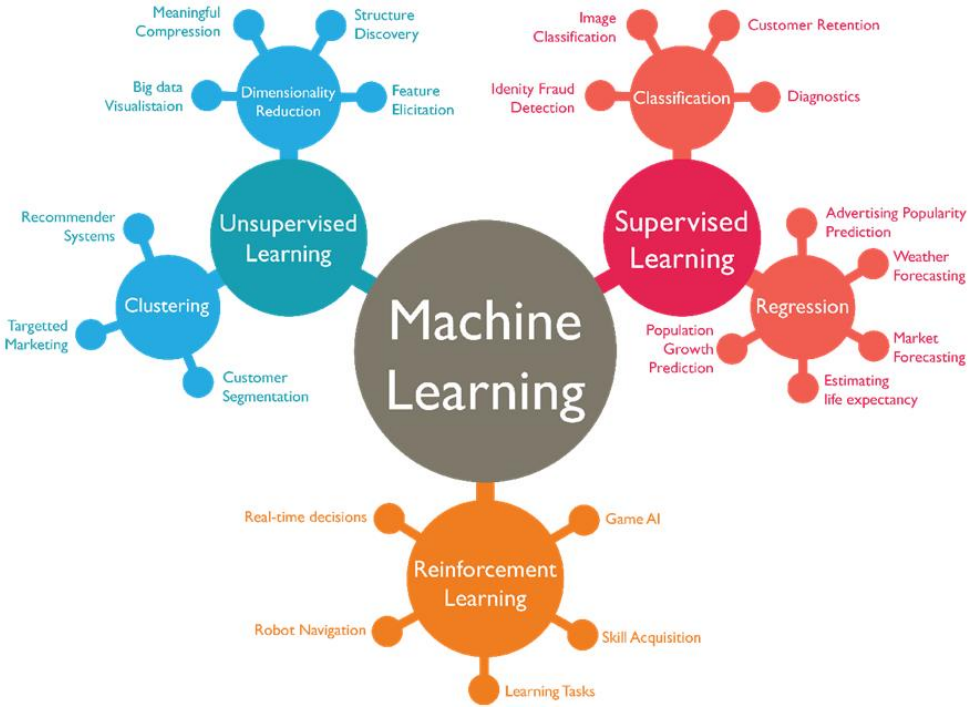

voice-assistant

## Let’s try ML…


Click here

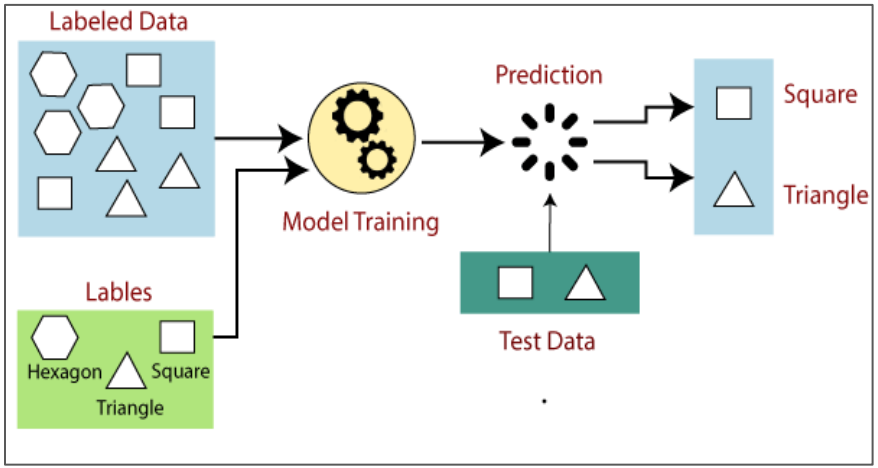
Reference link

## Types of Machine Learning

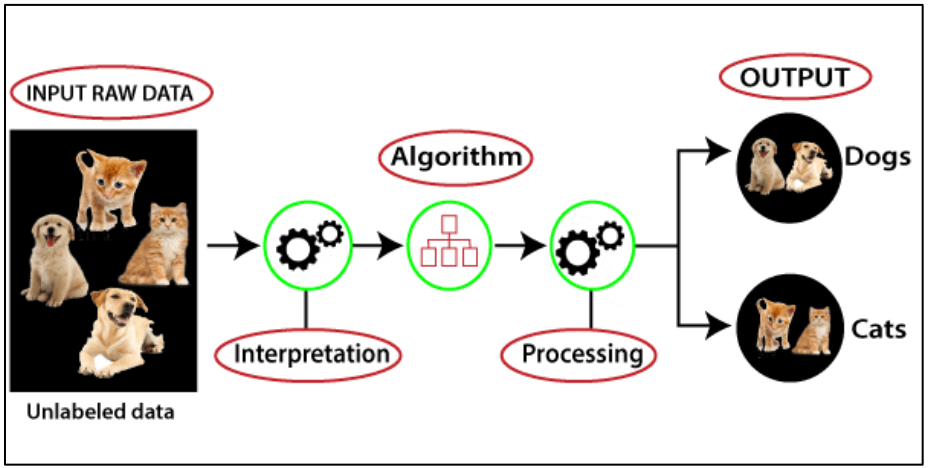



[Types of ML]

## ML Categories: Supervised & Unsupervised



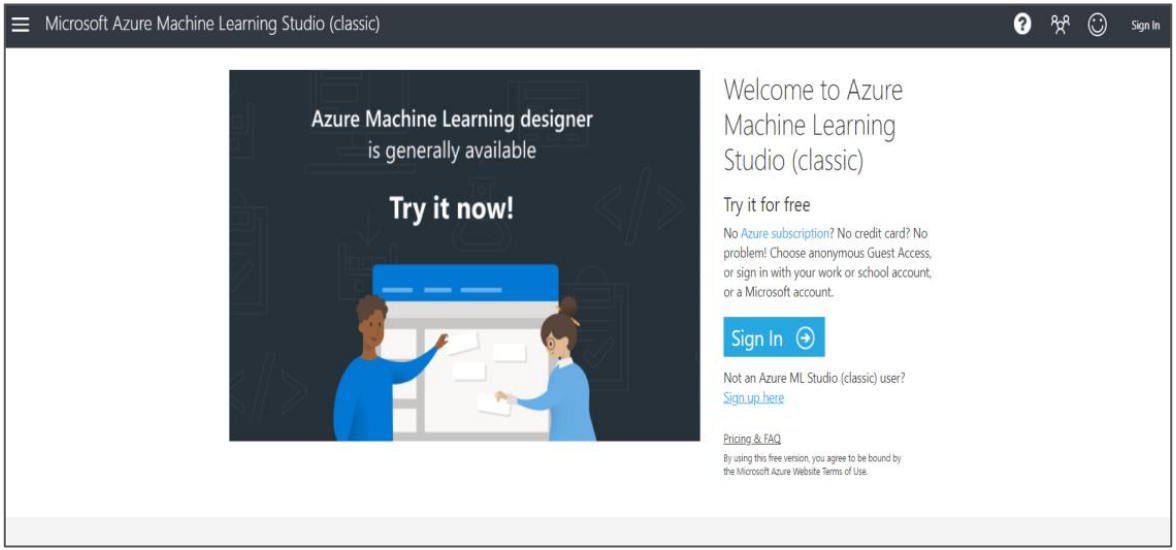

Supervised Learning



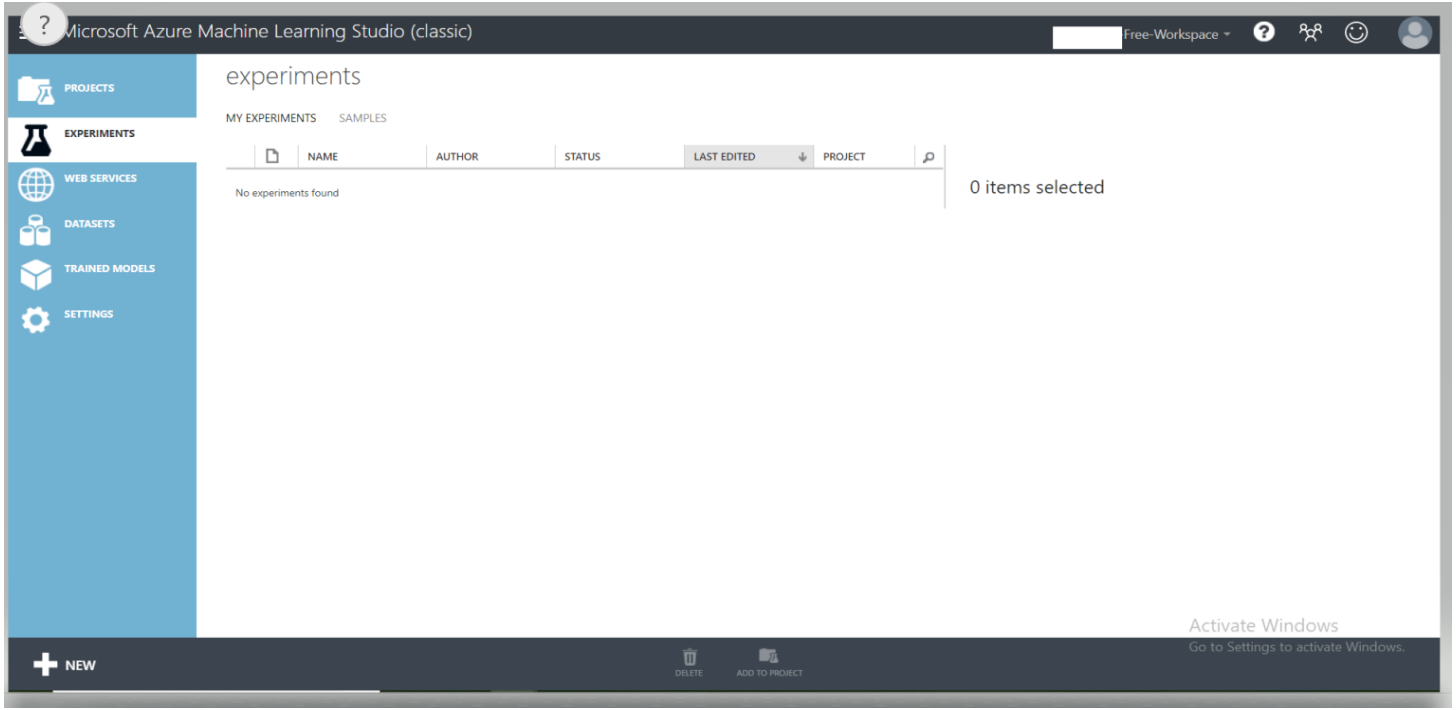

Unsupervised Learning

## ML Hands-on: GUI V/S Bare Coding

- Two approaches are there to practice ML.
- Dedicated Cloud services for GUI (eg:- https://studio.azureml.net/)
- Customized modeling needs bare coding.

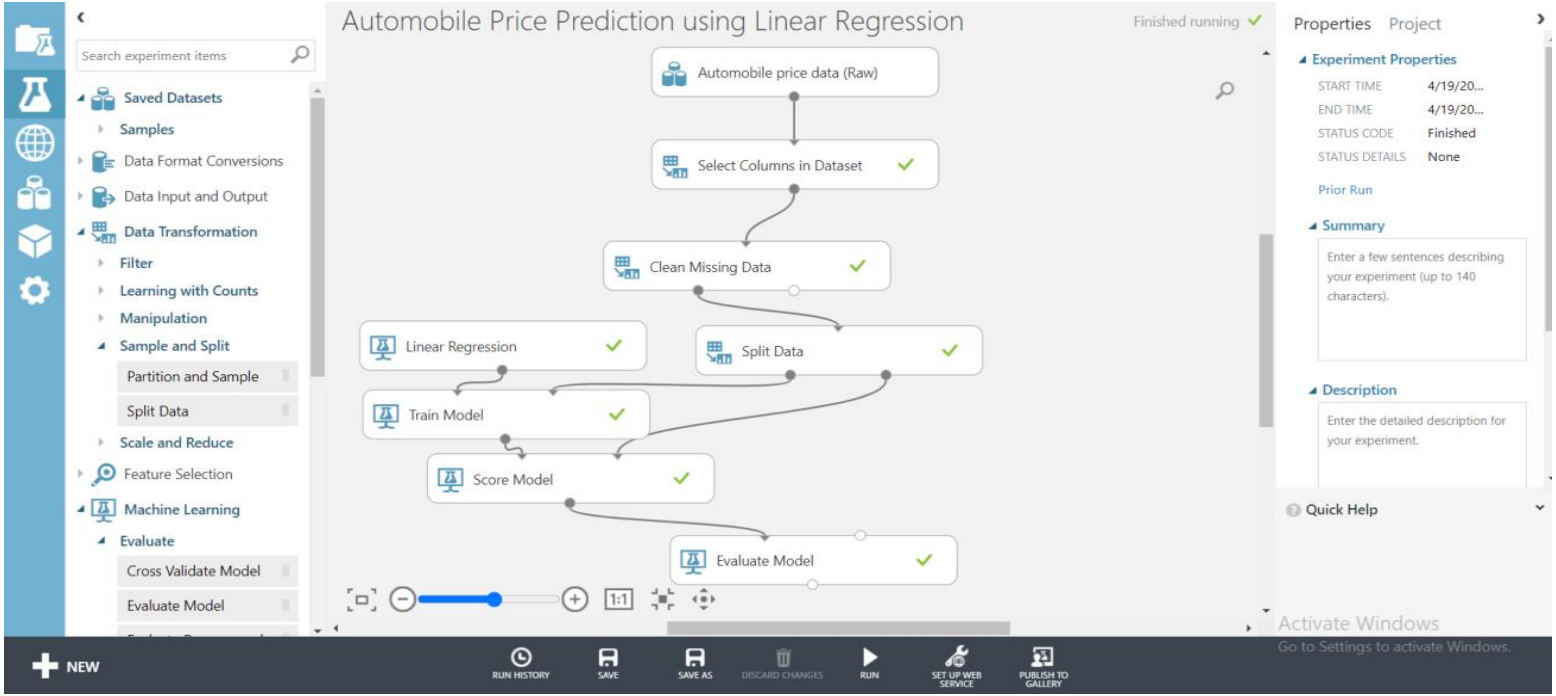


## Microsoft Azure Workspace




https://studio.azureml.net/

## Model Development with Azure ML Studio




https://studio.azureml.net/

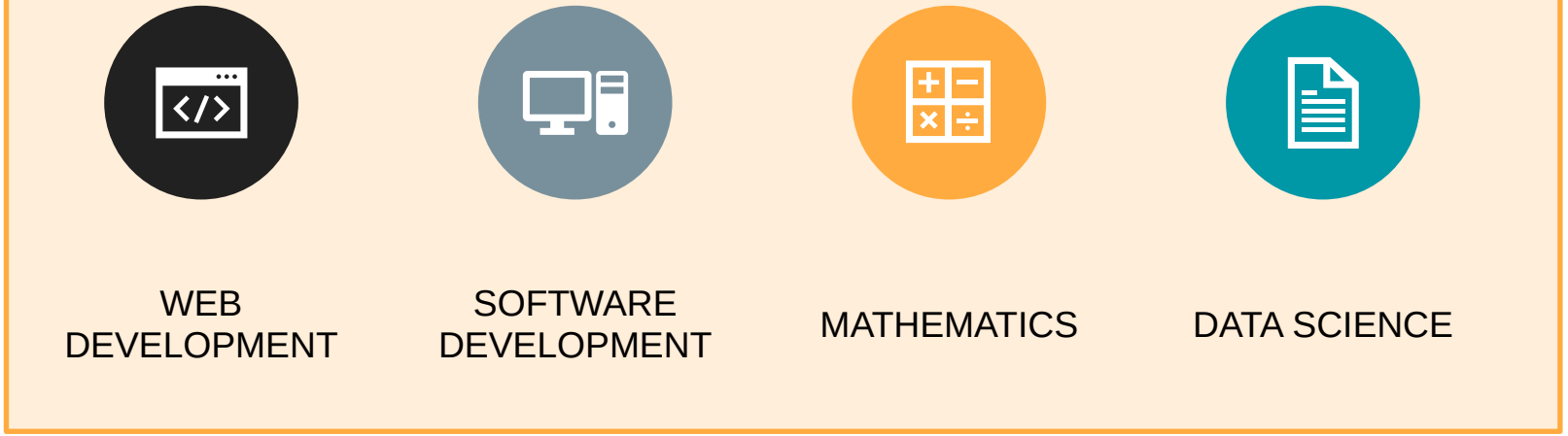
## Coding Platform: Python

- Python is a General-Purpose Programming language that is often applied in scripting roles.
- So, Python is programming language as well as scripting language.
- Python is an Interpreted language

python™

Python

## Python - Uses




Python Applications

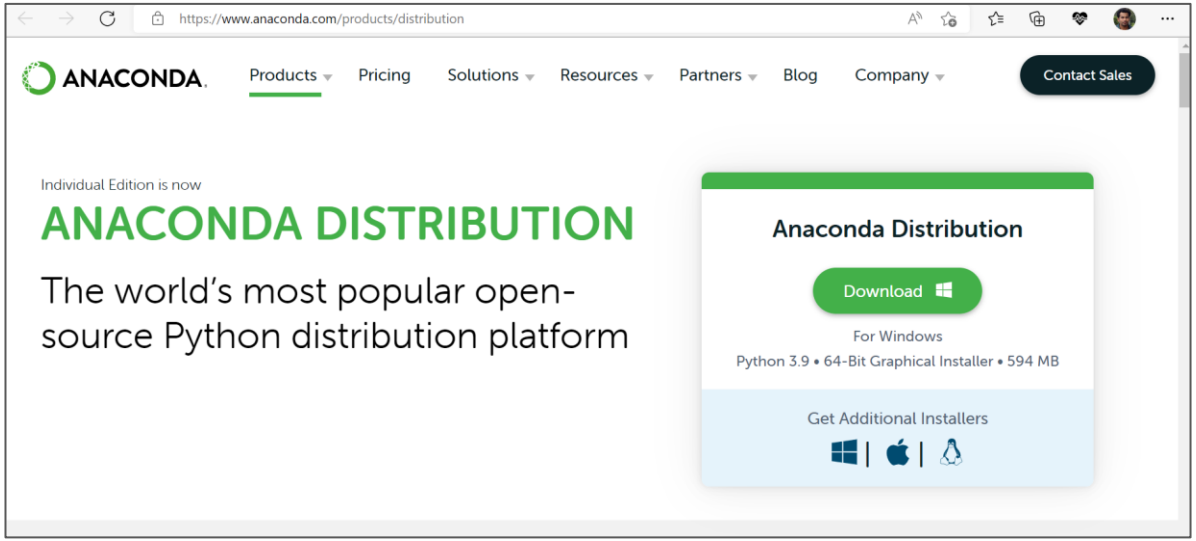
## Anaconda : Introduction

- An easy-to-install collection of high-performance Python libraries
- For managing packages and environments.
- Can use Conda to install over 1.5k packages (including the R language) from the Anaconda public repository.
- More than 20k packages from community channels, such as Conda-forge and bioconda

**Lab 1 -** Demonstration of Anaconda Installation

## Anaconda Installation

- Visit the following link: https://www.anaconda.com/distribution/
- Scroll down the page and select windows.




Anaconda

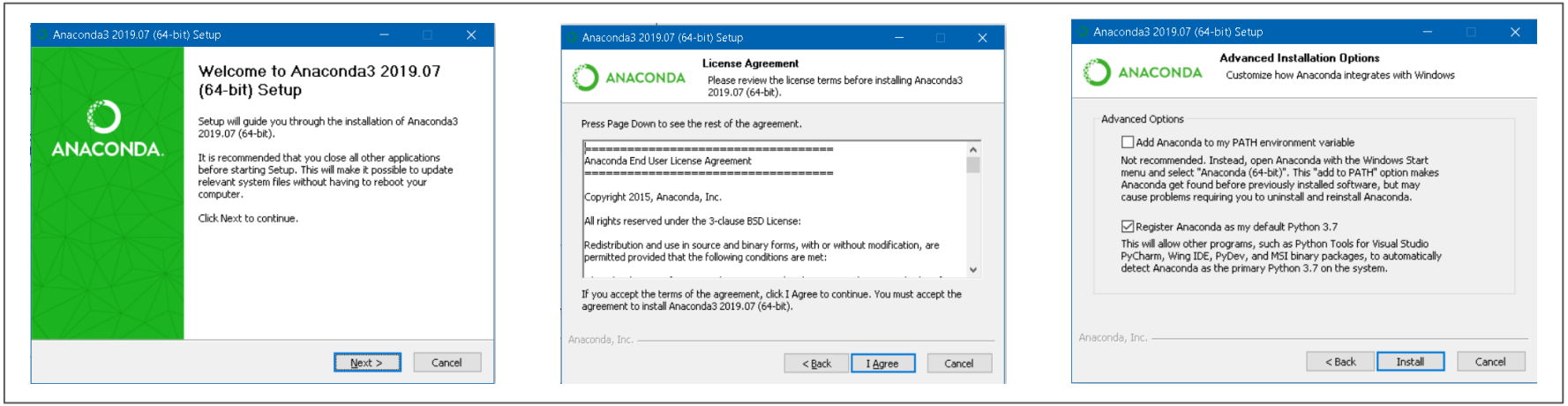
## Anaconda Installation….

- Download version matching to your machine


https://www.anaconda.com/download/

## Anaconda Installation….

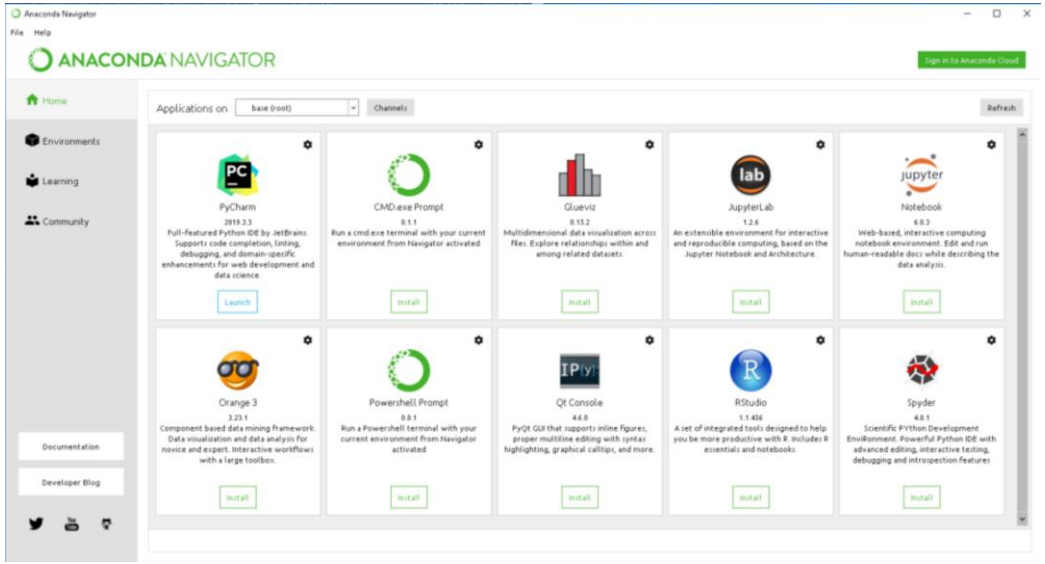
- Run the installation file and accept product terms




https://docs.anaconda.com/free/anaconda/install/windows/

## Anaconda Navigator

- Anaconda Navigator is a desktop graphical user interface included in Anaconda that allows you to launch applications and easily manage conda packages, environments and channels without the need to use command line commands.

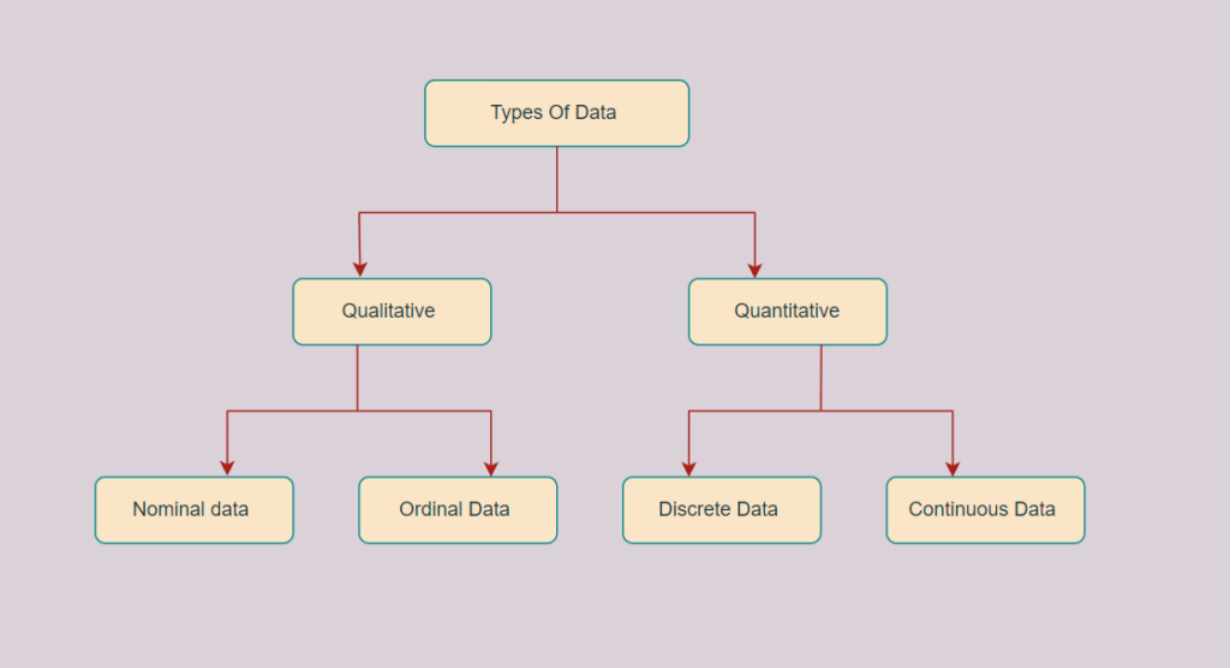



https://anaconda.org/anaconda/anaconda-navigator

## Types of Data

**Example**

- Nominal – Good, Bad,…
- Ordinal – First, Second….
- Discrete – Student count
- Continuous -- Temperature




Reference

## Requisite libraries: Numpy, Pandas & Seaborn

### Numpy

- Multidimensional arrays and matrices
- High-level mathematical functions
- **pip install numpy**

```
>>> import numpy
>>> numpy.__version__
```

### Pandas

- Easy data structure
- quicker data analysis
- Structed &  Un-structured data
- **pip install pandas**
- **import pandas as pd**

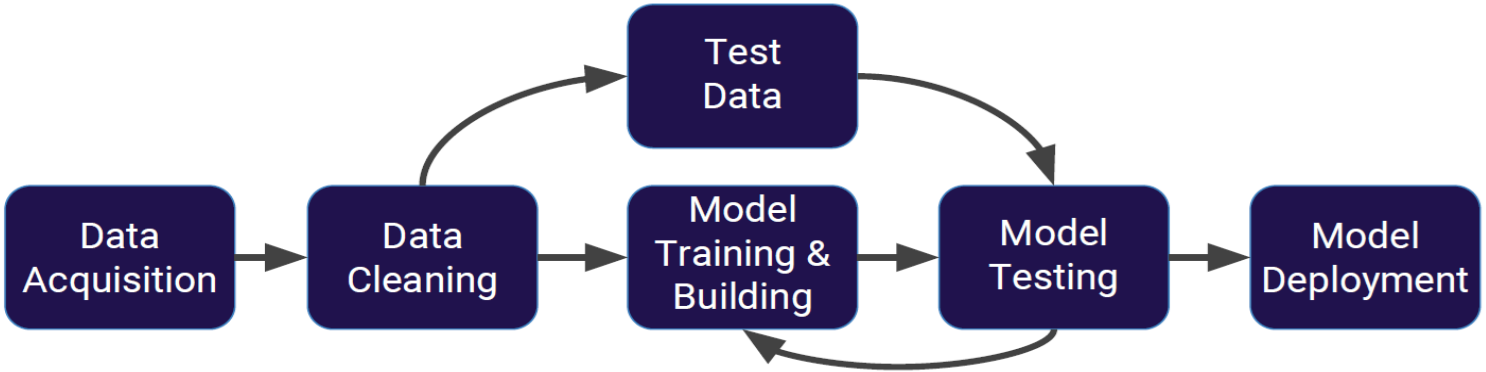
### Seaborn

- Seaborn
- Visualization library
- Statistical graphics plots
- Relational patterns
- Used for EDA
- **pip install seaborn**
- **import seaborn as sns**

## Scikit-Learn Library

- A free machine learning library.
- Includes most of the classification, regression and clustering algorithms.
- Works with numerical and scientific libraries, NumPy and SciPy.
- Machine learning pipelined tasks are already in scikit learn.
- Includes pre-processing, feature selection, data splitting, customize algorithms, fitting models, tuning parameters, prediction, evaluation, and exporting the model.
- Has an extended support for Deep learning and cloud services.

## Machine Learning Workflow



## Writing your first Machine Learning Codes in Only 6 Lines!

Write a code to differentiate between Apples & Oranges ?

## Training Data

**Features**

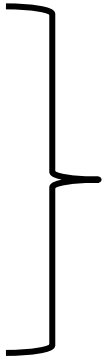
| Input of classifier | | Output of classifier |
|---|---|---|
| **Weight** | **Texture** | **Label** |
| 150g | Bumpy | Orange |
| 170g | Bumpy | Orange |
| 140g | Smooth | Apple |
| 130g | Smooth | Apple |
| … | … | … |

1. import sklearn
2. features = [[140,"smooth"],[130,"smooth"],[150,"bumpy"],[170,"bumpy"]]
3. labels = ["apples", "apples", "orange", "orange"]

Change strings to integers

## First 3 Lines of Code !

```
import sklearn
features = [[140, 1], [130, 1], [150, 0],
[170, 0]]
labels = [0, 0, 1, 1]
```

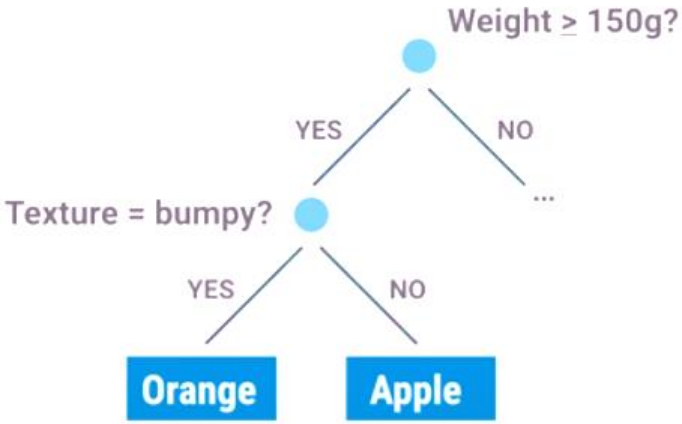


0: bumpy ; 1: smooth
0: apple ; 1: orange

## Classifier

- Box of RULES
- Learning Algorithms are the procedure that creates RULES, by finding patterns in your training DATA.
- Ex: It creates RULE that heavier fruit is more likely to be an orange!

## Decision Tree



## Machine Learning……Final Code

```
1. from sklearn import tree
2. features = [[140, 1], [130, 1], [150, 0], [170, 0]]
3. labels = [0, 0, 1, 1]
4. clf = tree.DecisionTreeClassifier()
5. clf = clf.fit(features, labels)
6. print(clf.predict([[150, 0]]))
```

0: bumpy ; 1: smooth

0: apple ; 1: orange

Classifier gets trained on input data

## Need for Model Evaluation

- Built on a subset of the total data, termed as training data, and they are used to predict on new data that is not part of this training subset.
- If a model is totally adapted to its training data, it would fail to predict accurately any new data (**Overfitting**).
- If model is too general, it would predict poorly on particular cases (**Underfitting**).
- A good model should be perfectly balanced to avoid **both**.
- By holding out part of the data from the training set and evaluating model with this subset of test data.

## ML: Problem Types

| Problem types | Algorithms |
|---|---|
| Regression | Linear regression, K-NN, random forest, neural networks |
| Classification | Logistic regression, random forest, K-NN, gradient boosting classifier, neural networks |
| Clustering | K-Means, DBSCAN, Hierarchical clustering, Gaussian mixture models, BIRCH |
| Time-series forecasting | ARIMA, SARIMA, LSTM, Exponential smoothing, Prophet, GARCH, TBATS, Dynamic linear models |
| Anomaly detection | IsolationForest, Minimum covariance determinant, Local outlier factor, One-class SVM |
| Recommendation | Content-based and collaborative filtering machine learning methods |
| Data generation | Generative adversarial network (GAN), Hidden Markov models |

# Git & GitHub



Reference

## Git

- **Git** is software for tracking changes.
- Handle any set of files
- Used for work among programmers collaboratively
- Developing source code during software development.

Click here

Reference link

## GitHub

- Provider of Internet hosting for software development and version control.
- Offers the distributed version control
- Supports source code management (SCM)


Click here

Reference link

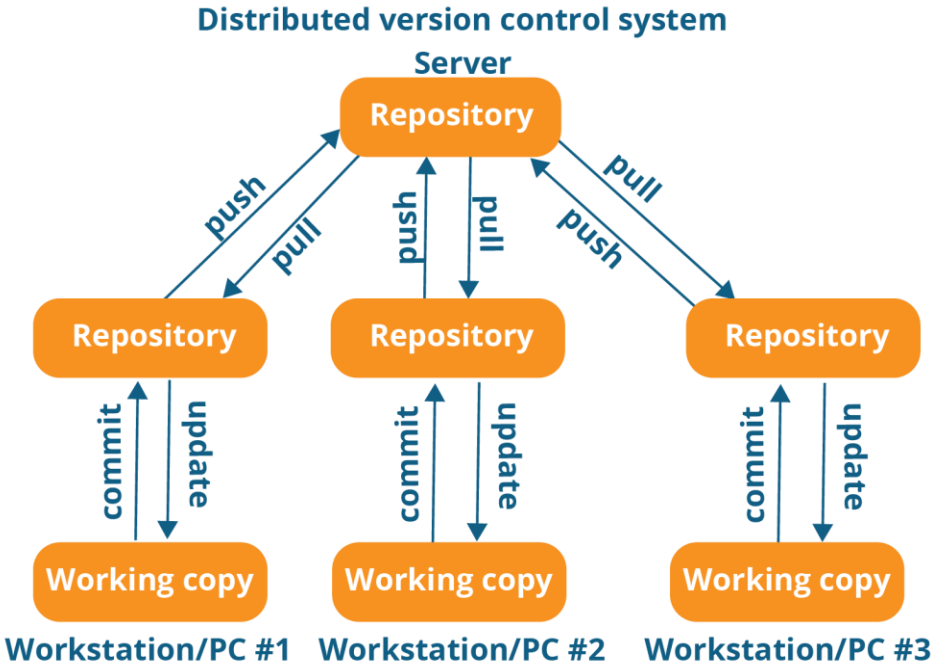
## Difference between Git and GitHub

| Git | GitHub |
|---|---|
| Installed locally | Hosted in cloud |
| First released in 2005 | Company launched in 2008 |
| Maintained by The Linux Foundation | Purchase in 2018 by Microsoft |
| Focused on version control and code sharing | Focused on centralized source code hosting |
| Primarily a command-line tool | Administered through the web |
| No user management features | Built-in user management |
| Minimal external tool configuration features | Active marketplace for tool integration |
| Competes with Mercurial, Subversion, IBM, Rational Team Concert and ClearCase | Competes with Atlassian Bitbucket and GitLab |
| Open source licensed | Includes a free tier and pay-for-use tiers |

## Version Controlling

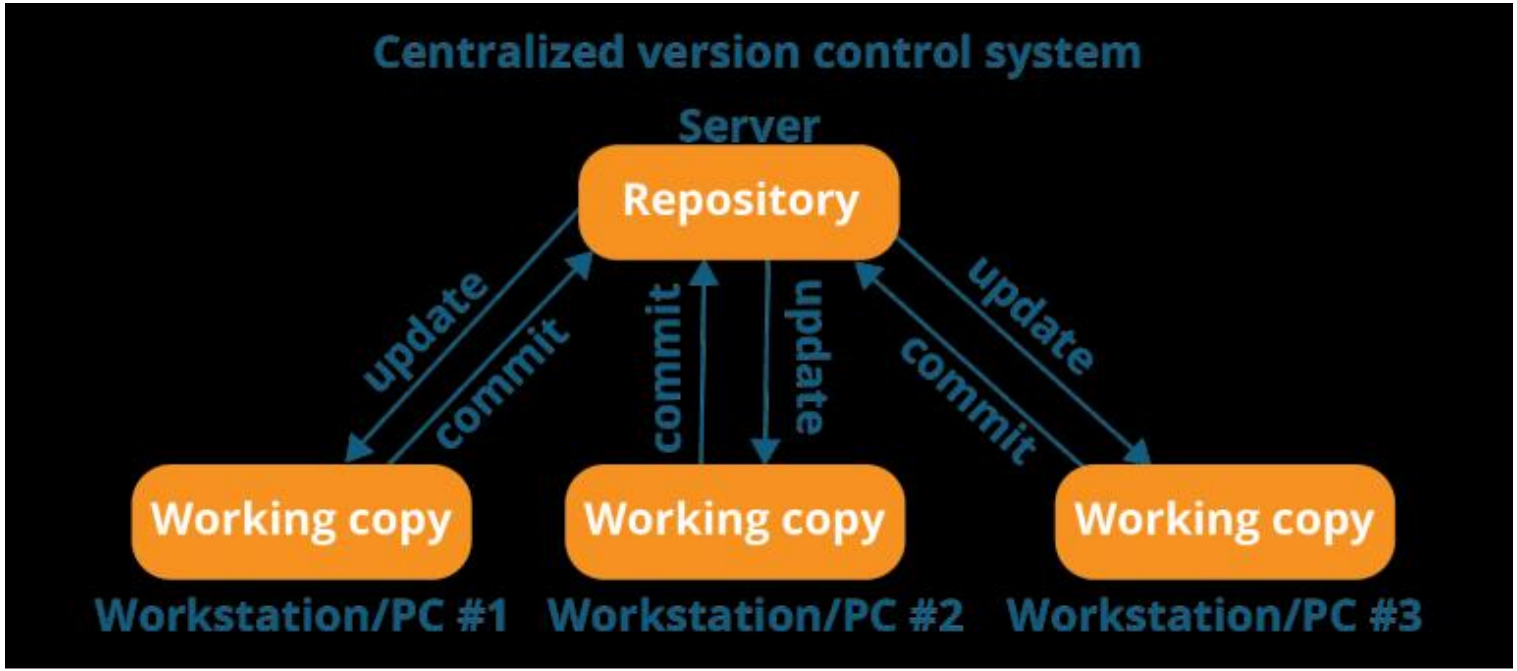
- Version control (also known as revision control, source control, or source code management)
- Responsible for managing changes to computer programs.
- Handle documents, large web sites, or other collections of information.

## Distributed Version Control



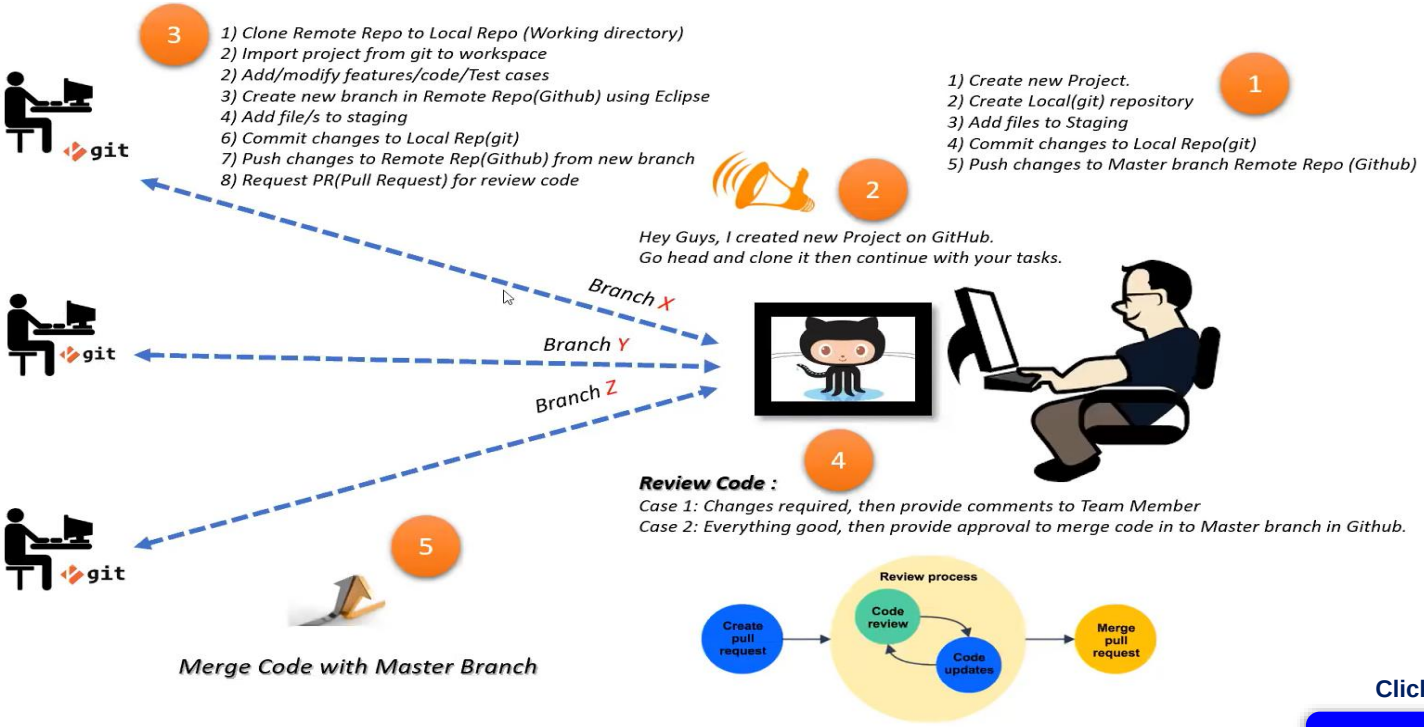

Distributed control

## Centralized Version Control




Centralised control

## Code-Cycle




Click here

Reference link

## Git CMD Vs Git bash

- Git CMD just like Windows CMD.
- Can call all Git features

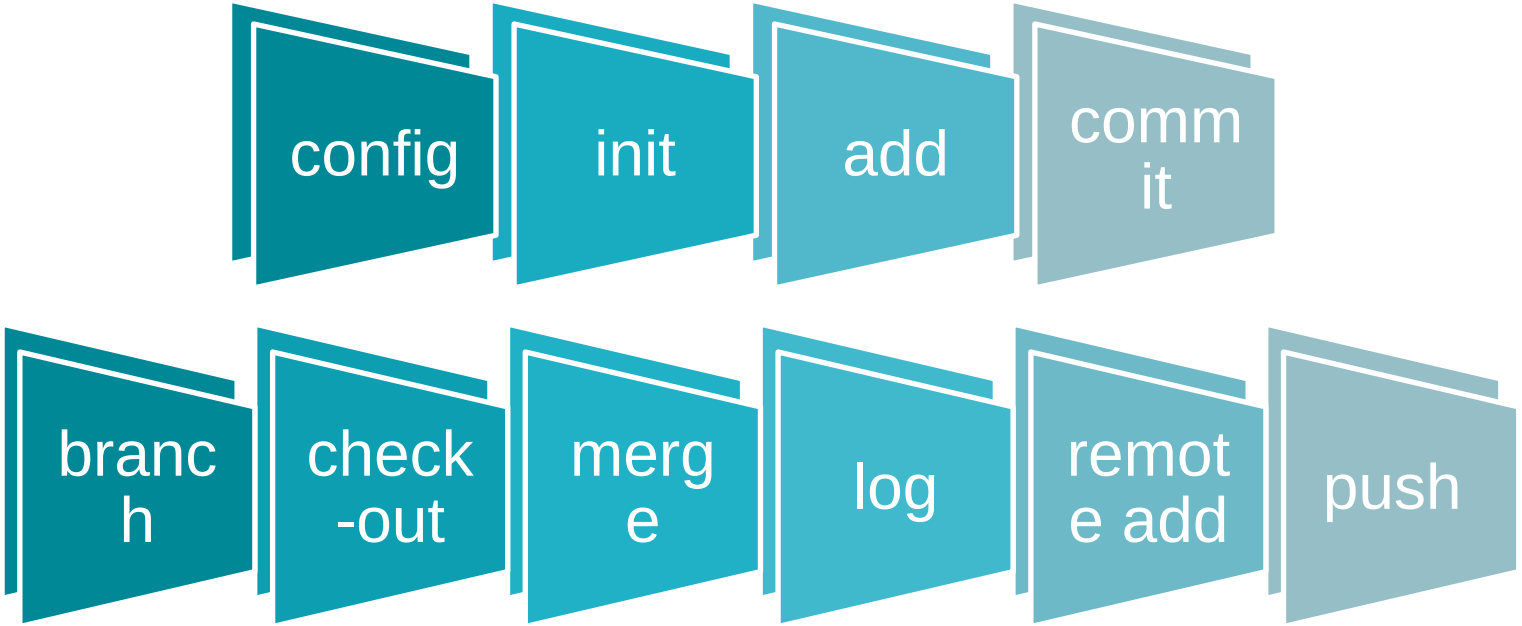

- Git Bash emulates bash environment
- Also support Unix commands


Click here

Reference link

## Lab 2 - GitHub Commands

## Git Commands



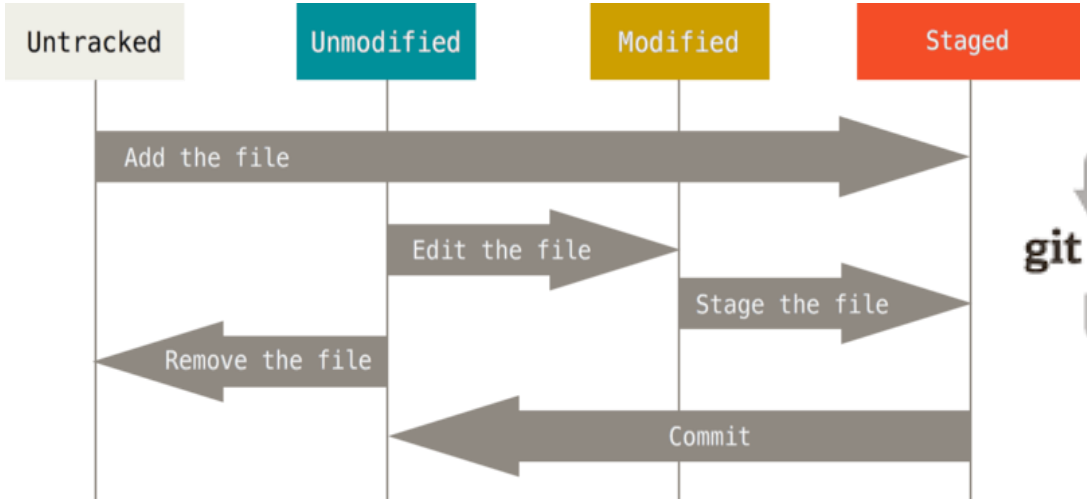

Click here

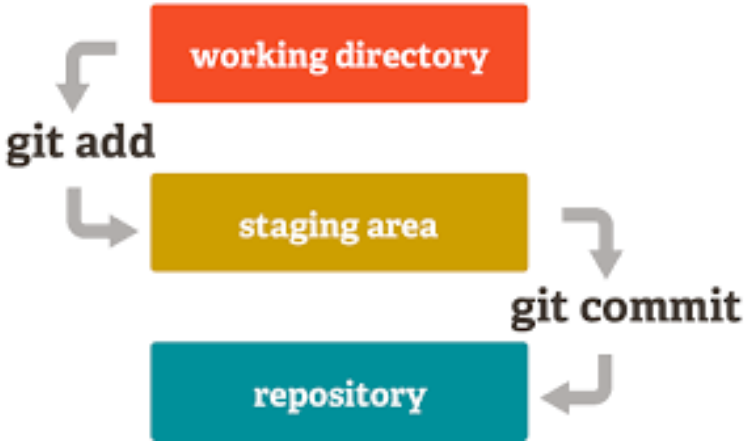
Reference link

## Processing Stages

- **Untracked:** Files which are newly created in working directory and git does not aware of these files.
- **Staged:** Files which are added to staging area. These files are ready to commit.
- **Committed:** Files which is committed and placed in local repository/ Committed State.
- **Modified:** File which is already tracked by git. But is modified in working directory .

## Git-Flow



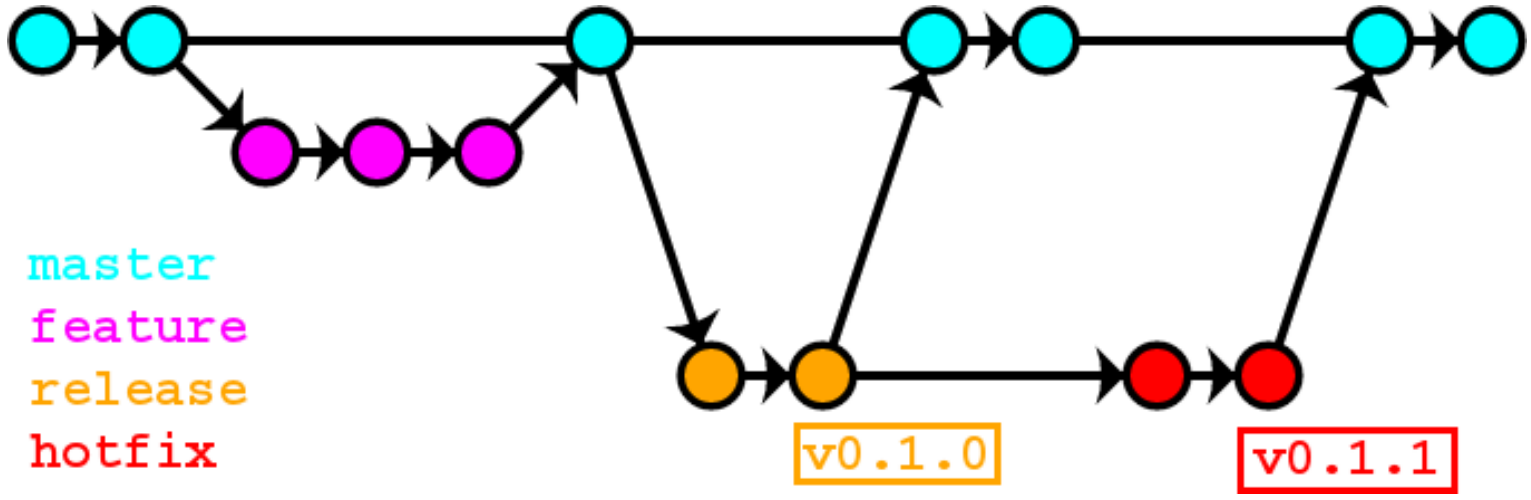




Staging Tree

Untracked file – Red color

Staging Area – Green Color

## Branching




Master and branch

## Create Account on GitHub

https://github.com/join?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home

## Lab 3 - Getting started with the GitHub


Connect to GitHub
GitHub
Sign in
Sign in with your browser
or
Personal Access Token
Sign in
Don't have an account? Sign up

GitHub

## Summary

- In the first section, we see definition of Machine Learning, that enables computers to learn from data. We delve into key types like Supervised, Unsupervised, and Reinforcement Learning, showcasing their real-world applications and examining different Machine Learning products.

- The second part focuses on Git and GitHub. Git is a version control system, while GitHub is a collaborative platform. We explain their roles in tracking changes, facilitating teamwork, and managing conflicts. Towards the end, we guide you through creating a GitHub account.

- Emphasizing best practices for version control, we highlight the benefits of using Git and GitHub for efficient collaboration and code management.

## 1. Which type of Machine Learning involves learning from a labeled dataset to make

a) Unsupervised Learning

b) Supervised Learning

c) Semi-Supervised Learning

d) Reinforcement Learning

**B) Supervised Learning**

2. **Which type of Machine Learning Library use for Prediction of based on past data**

a) Pandas

b) Numpy

c) Sci-kit Learn (skleran)

d) CV

C) Sci-kit Learn (skleran)

## 3. Git is _____ Version Control system

a) Distributed

b) Centralized

**Answer: a) Distributed**

## 4. What is the main goal of Unsupervised Learning?

a) To make predictions based on labeled data

b) To learn from rewards and punishments

c) To find hidden patterns and structures in unlabeled data

d) To improve the accuracy of existing models

Answer - c) To find hidden patterns and structures in unlabeled data

## 5. Which Python Library Use for Data Visualization

a) lineplot

b) Seaborn

c) matplotlib

d) scatterplot

**Answer – b and c**

## References


- https://en.wikipedia.org/wiki/Git
- https://en.wikipedia.org/wiki/GitHub
- https://medium.com/machine-learning-101
- https://medium.com/@randylaosat/a-beginners-guide-to-machine-learning-dfadc19f6caf
- https://sml.csa.iisc.ac.in/Courses/Spring21/E0_270/pdfs/1.pdf
- https://www.geeksforgeeks.org/difference-between-git-and-github/
- https://link.springer.com/book/10.1007/978-3-030-81935-4
- https://ai.stanford.edu/~nilsson/MLBOOK.pdf
- https://mitpress.mit.edu/9780262043793/introduction-to-machine-learning/

Thank you...!