Experiment 1

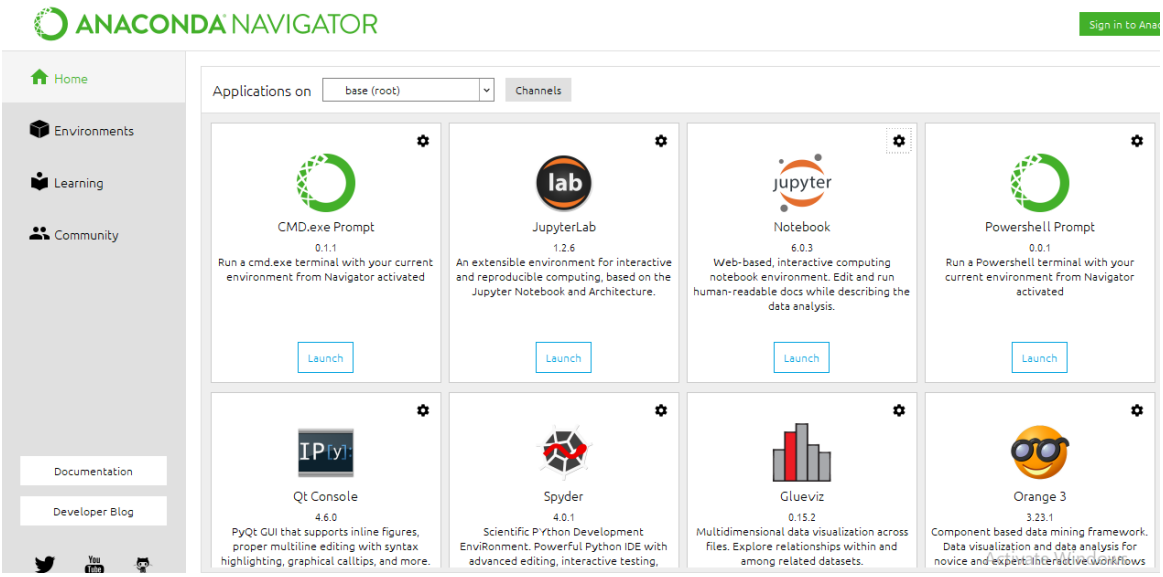
Software setup of machine learning environment

Name:-Shivansh Srivastava

AdmNO:-18SCSE1010734

Install Anaconda:-

Anaconda is a package manager, an environment manager, and Python distribution that contains a collection of many open source packages. This is advantageous as when you are working on a data science project, you will find that you need many different packages (numpy, scikit-learn, scipy, pandas to name a few), which an installation of Anaconda comes preinstalled with. If you need additional packages after installing Anaconda, you can use Anaconda's package manager, conda, or pip to install those packages. This is highly advantageous as you don't have to manage dependencies between multiple packages yourself. Conda even makes it easy to switch between Python 2 and 3 (you can learn more here). In fact, an installation of Anaconda is also the recommended way to install Jupyter Notebooks



Configure Git:-

Git is the most commonly used version control system. Git tracks the changes you make to files, so you have a record of what has been done, and you can revert to specific versions should you ever need to. Git also makes collaboration easier, allowing changes by multiple people to all be merged into one source.

The git config command is a convenience function that is used to set Git configuration values on a global or local project level. These configuration levels correspond to .gitconfig text files. Executing git config will modify a configuration text file. We'll be covering common configuration settings like email, username, and editor. We'll discuss Git aliases, which allow you to create shortcuts for frequently used Git operations. Becoming familiar with git config and the various Git configuration settings will help you create a powerful, customized Git workflow.

Usage

The most basic use case for git config is to invoke it with a configuration name, which will display the set value at that name. Configuration names are dot delimited strings composed of a 'section' and a 'key' based on their hierarchy. For example: user.email

➔ git config user.email

In this example, email is a child property of the user configuration block. This will return the configured email address, if any, that Git will associate with locally created commits.

git config levels and files

Before we further discuss git config usage, let's take a moment to cover configuration levels. The git config command can accept arguments to specify which configuration level to operate on. The following configuration levels are available:

--local

By default, git config will write to a local level if no configuration option is passed. Local level configuration is applied to the context repository git config gets invoked in. Local configuration values are stored in a file that can be found in the repo's .git directory: .git/config

--global

Global level configuration is user-specific, meaning it is applied to an operating system user. Global configuration values are stored in a file that is located in a user's home directory. ~ /.gitconfig on unix systems and C:\Users\<username>\.gitconfig on windows

--system

System-level configuration is applied across an entire machine. This covers all users on an operating system and all repos. The system level configuration file lives in a gitconfig file off the system root path. $(prefix)/etc/gitconfig on unix systems. On windows this file can be found at C:\Documents and Settings\All Users\Application Data\Git\config on Windows XP, and in C:\ProgramData\Git\config on Windows Vista and newer.

Thus the order of priority for configuration levels is: local, global, system. This means when looking for a configuration value, Git will start at the local level and bubble up to the system level.

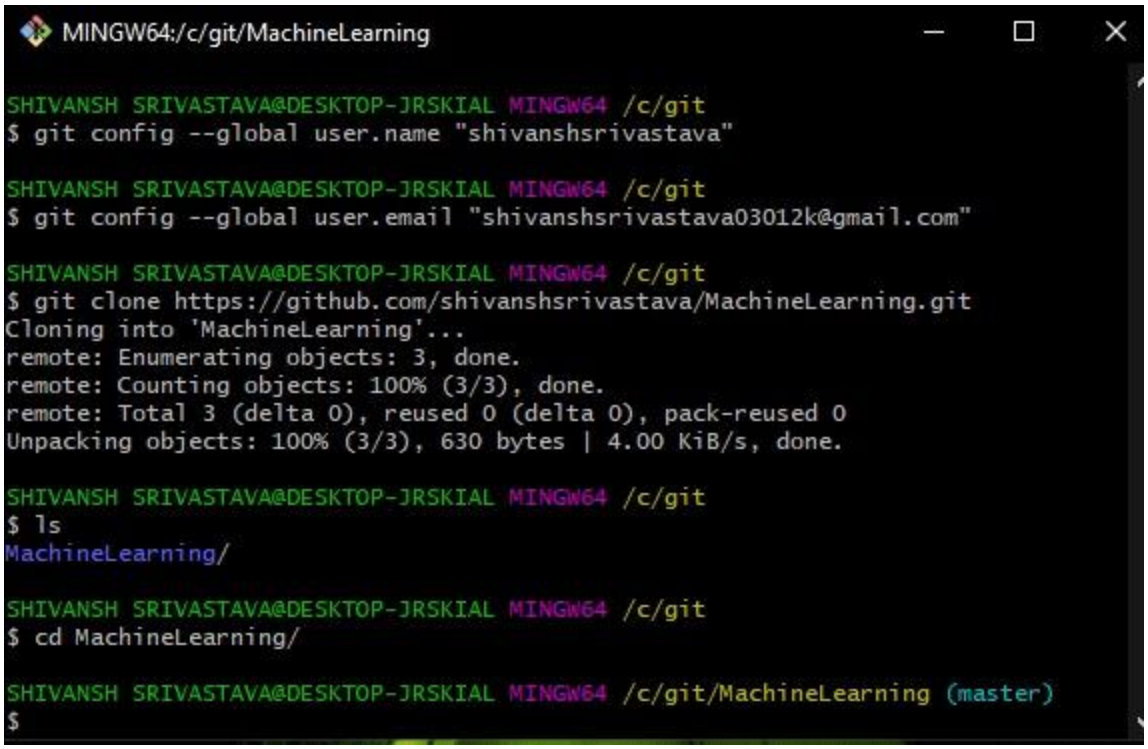
Writing a value

Expanding on what we already know about git config, let's look at an example in which we write a value:

➔ git config --global user.email your_email@example.com
➔

This example writes the value your_email@example.com to the configuration name user.email. It uses the --global flag so this value is set for the current operating system user.

and connect with GitHub



GitHub:-

GitHub is a website and cloud-based service that helps developers store and manage their code, as well as track and control changes to their code

MY GitHub Link:-

https://github.com/shivanshsrivastava