What is Docker and Kubernetes?

If these terms are new to you, please refer to these resources.

- A brief introduction of docker (video): https://www.youtube.com/watch?v=rmf04ylI2K0
- A beirf intorduction of docker (blog): https://opensource.com/resources/what-docker
- A brief introduction of kubernetes (video): https://www.youtube.com/watch?v=R-3dfURb2hA
- A brief introduction of kubernetes (documentation): https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/

Once the account is created, a user will get login details of a login node. The login node has Ubuntu 18.04 OS installed, with neither GPU support nor any computational tools such as TensorFlow or Pytorch installed. The user is requested to copy all the necessary code/data to the login node and submit a CPU/GPU job to kubernetes cluster using the instructions mentioned below.

How to submit a job request to Kubernetes cluster

Job description file

Create a job file for the task that you need to execute on the cluster. A sample job file is given below:

```
apiVersion: v1
kind: Pod
metadata:
  # Change the name according to your need. A good name is descriptive and should tells information about what kind of experiment the pod will be used for.
  name: name_of_the_pod
spec:
  securityContext:
    # Replace NUMBER with the user id provided by the administrator
    runAsUser: NUMBER
    fsGroup: NUMBER
  containers:
```

```
    # name of the container running inside the pod. No need to update this
  - name: tf-container
    # Name of the docker image you want to run. The Kubernetes will pull the image from docker hub. The following image has tensorflow and some
other python tools installed. This image is created using the Dockerfile available at: https://github.com/abhipec/iitg_kubernetes . According to your
requirement, create the necessary image.
    # iitg_kubernetes:p3_master_image_latest is the name of the image where several Machine Learning realted python 3 packages are already
installed.
    # iitg_kubernetes:p2_master_image_latest is the name of the image where several Machine Learning related python 2 packages are already
installed.
    image: abhipec/iitg_kubernetes:p3_master_image_latest
    # Once the container is started, the following command will be run. Note that all the data moved to the login node home directory will be
accessible to the container in the /data/ directory.
    command: [ "/bin/bash", "/data/what_ever_code_you_want_to_run" ]
    # For demo, to check working of kubernetes, comment the above command and uncomment the following commands.
    # command: [ "/bin/sleep", "120" ]
    # The above command will create the container, then run it for 120 seconds and then terminate.
    # If let say, you need to run a bash script, then the command woud be
    # command: [ "/bin/bash", "/data/name_of_script.bash" ]
    # Request the number of GPU nodes.
    resources:
     limits:
       nvidia.com/gpu: 1
    volumeMounts:
     - name: nfs-data
       mountPath: /data
  volumes:
  - name: nfs-data
   hostPath:
     # Replace the USERNAME with your login account name.
     path: /home/kubernetes/data/USERNAME/
     type: Directory
  # Do not change the below mentioned parameter.
  restartPolicy: Never
```

Submitting a job request

Submit the job using the following command:

```
kubectl create -f job_description_file.yaml
```

Monitoring job progress and debugging

Job status

To check the status of the job, use the following command:

```
kubectl get pods
```

The status messages have the following meaning:

Status	Meaning
Running	The job is currently running.
CreatingContainer	The Kubernetes cluster is creating the docker container. It might take some time, especially when a new docker image is pulled from the docker hub site.
Error	The job terminated with an error
Completed	The job has successfully completed
Pending	The scheduler is waiting for a node which has the requested resources available.
ImagePullBackOff	It occurs when their is an issue in pulling the specified docker image either due to internet connectivity or wrong image name.

Monitoring job

If the job in execution prints output to stdout, then to view that output, you can use the kubectl logs command mentioned below. This will continuously display the stdout messages on the terminal. Also if the job is terminated with the Error status, detailed information of the error will be provided by this command.

```
kubectl logs -k pod/name_of_the_pod_in_job_description_file
```

Also, if the job is with the running status, it is possible to attach the shell withing the container to your terminal with the following command.

```
kubectl exec -it pod_name bash
```

Pending jobs

To know why a job is in pending status, use the following command:

```
kubectl describe pod name_of_the_pod
```