

Running Knn GPU code on Amazon AWS

Pseudo code

Inputs: Train Data D , Test Data X , Number of nearest neighbors k

Output: Predicted class labels of X

Step 1: Copy D and X to the GPU shared memory. Split D into threads

Step 2: Kernel 1: Calculate the distance between each $d_i \in D$ and each $x_j \in X$

Step 3: Send the distances to CPU

Step 4: Copy distances to GPU shared memory, split into threads

Step 5: Kernel 2: Sort distances for each test instance

Step 6: Send indices of k nearest neighbors to CPU

Step 7: CPU: Assign most frequent class label from nearest neighbors as predicted class label

How to run the code

Step 1: Logon to Amazon AWS and under Services select 'Ec2'. Click on Launch Instance button.

Step 2: Under the Quick Start column on the left, click on Community AMIs

Step 3: Search for "ubuntu hvm 12.04", select the first AMI "**ubuntu/images/hvm/ubuntu-precise-12.04-
amd64-server-20130204** - ami-0745d26e"

Step 4: From the Instance types, select "GPU instances" and click on "Next: Configure Instance Details"; on the bottom right.

Step 5: Next is 'Configure Instance' step, we do not want to make any changes to that so, click on "Next: Add Storage" button on the bottom right.

Step 6: You can change the Size of the instance based on your need. The default is 8Gib, we changed it to 20 Gib. Click on "Next: Tag instance"; on bottom right.

Step 7: This is just to give a name to this instance, for now, give any name like '**GPU_instance**'. Click "Next: Configure Security Group"; on bottom right.

Step 8: Next comes the 'Configure Security Group', keep the default settings and click "Next: Review and Launch".

Step 9: Just check that what we configured is showing up. Click on "launch".

Step 10: In the 'Select existing Key pair or create a new key pair' dialog box, select create new key pair and name it. We gave the name 'dmkd_gpu'. This key pair file will be used to login to the instance. Acknowledge and click on Launch instance.

Step 11: If everything went well Launch Status page will show up with "Your instance is now launching" statement. Click on "View Instance" button on bottom right. You could see you instance in the running instances by the name '**GPU_instance**'.

Step 12: When you select the instance, Instance description shows up at the bottom. Copy the 'Public DNS' value, it will be something like 'ec2-.....-amazonaws.com'.

Windows Login

Step 1: Open [Puttygen](#) and in the dialogbox click on 'Load' and select the keypair file which you downloaded in Step 10. In the search dialog box, select 'All files' at the bottom right which will show the '.pem' file. Click on open and click 'OK' for successful import notice. Click on 'Save private key' at the bottom right and click 'yes' to the warning. Save the file with the same name and without the '.pem' extension, Putty will automatically add the .ppk extension to the newly created file.

Step 2: Open Putty and in the 'Host name' type 'ubuntu@<DNS value which you copied in Step 12>'. Like 'ubuntu@ec2-.....-amazonaws.com'.

Step 3: Under the connection category in the left panel, select the '+' near 'SSH' and click on 'Auth'. Browse for the '.ppk' which we created in Step 12 and click open. If everything goes well, it will connect to the amazon instance.

Copying Code

Step 1: Open [WinSCP](#) and in the dialog box for 'Host name', paste the DNS name which you copied in Step 12 of instance creation. Under User name, type "ubuntu" and click on 'Advanced' under Password text box.

Step 2: In the 'Advanced Site Settings' pop up, click on 'Authentication' under 'SSH' in the left column. Browse for the Private Key file '.ppk' which you created in Step 1 of 'Windows Login'.

Step 3: Click on 'Login' in the main window and it will connect you the amazon AWS instance.

Step 4: Browse the GPU code in the left window and copy the folder on the right side (on instance, /home/ubuntu).

Running the Code

Create a folder "Knn" on Aws instance. Unzip the file GPU_Knn.zip and upload everything to 'Knn' folder through WinSCP.

Step 1: Connect to the instance and browse to the 'Knn' directory and type the following

- `sudo dpkg -icuda-repo-ubuntu1204_5.5-0_amd64.deb`

Step 2: Type

- `sudo apt-get update`
- `sudo apt-get install cuda`

Wait for the installation completion. For more details on installation click [here](#) and follow the first three steps.

Step 3: Type

- `./test_knn.o 20 5 10 4`

to run the code. The first arguments are following

- `./test_knn.o` – executable file
- 20 – number of train instances
- 5 – number of test instances
- 10 – number of columns
- 4 – number of nearest neighbor to consider

After running k-nn algorithm, it will print the accuracy and time taken in milliseconds.

Note: You have to provide train data, test data, train labels, test labels as separate text files. All these files have to be present in the same directory.

Cleanup (Important)

Step 1: Logon to Amazon AWS and under Services select 'Ec2'.

Step 2: Under the 'Instances' tab in the left column; click on 'Instances'.

Step 3: Locate your instance and select it. On the top locate 'Actions' drop down button and click 'Stop' to stop instance. You can start it and connect to the same settings whenever you want. If you terminate it, you have to create a new instance all together.