TABLE OF CONTENTS

Chapter 1. Introduction to Using NGC with Azure ........................................................ 1
Chapter 2. Before You Start .......................................................................................... 2
  2.1. Prerequisites .................................................................................................. 2
  2.2. Setting Up SSH Keys ................................................................................... 3
  2.3. Setting Up a Security Group ......................................................................... 3
Chapter 3. Launching an NVIDIA GPU Cloud VM from the Azure Portal ......................... 4
  3.1. Creating Your GPU Cloud VM ................................................................. 4
  3.2. Connecting to Your GPU Cloud VM Instance .............................................. 8
  3.3. Starting and Stopping Your GPU Cloud VM ............................................. 8
  3.4. Deleting Your GPU Cloud VM and Associated Resources ......................... 8
Chapter 4. Launching an NVIDIA GPU Cloud VM Using Azure CLI .................................. 11
  4.1. Installing Azure CLI ............................................................................... 11
  4.2. Preparing Your VM Variables ................................................................ 11
  4.3. Creating Your GPU Cloud VM .............................................................. 12
  4.4. Connecting to Your GPU Instance with SSH .......................................... 13
  4.5. Stopping (Deallocating) and Starting VMs with the CLI ......................... 13
  4.6. Deleting VMs and Associated Resources with the CLI .......................... 14
Chapter 5. Using Premium Storage SSDs for Datasets .................................................. 16
  5.1. Creating Data Disk Storage Using the Azure Dashboard ......................... 16
  5.2. Creating Data Disk Storage Using the Azure CLI .................................... 17
  5.3. Mounting a Data Disk ............................................................................ 18
  5.4. Deleting a Data Disk ............................................................................ 18
Chapter 6. Using File Storage for Datasets .................................................................. 20
  6.1. Creating a Storage Account ................................................................ 20
  6.2. Creating File Storage ........................................................................... 21
  6.3. Mounting the File Storage .................................................................. 21
  6.4. Deleting File Storage .......................................................................... 22
Chapter 7. Examples of Running Containers ................................................................ 24
  7.1. Logging Into the NGC Container Registry ............................................. 24
  7.2. Running a Container .......................................................................... 24
  7.3. Example: MNIST Training Run Using TensorFlow Container ................ 25
Chapter 1.
INTRODUCTION TO USING NGC WITH AZURE

NVIDIA makes available on the Microsoft Azure platform a customized machine image based on the NVIDIA® Tesla Volta™ and Pascal™ GPUs. Running NGC containers on this virtual machine (VM) instance provides optimum performance for deep learning, machine learning, and HPC jobs.

Three flavors of the NVIDIA GPU Cloud image are available:

- Standard NVIDIA GPU Cloud Image
  Includes Ubuntu Server, the NVIDIA driver, Docker CE, and the NVIDIA Container Runtime for Docker
- GPU Accelerated Image for TensorFlow
  The standard image plus a built-in, ready-to-use TensorFlow container
- GPU Accelerated Image for PyTorch
  The standard image plus a built-in, ready-to-use PyTorch container

For those familiar with the Azure platform, the process of launching the instance is as simple as logging into Azure, selecting the NVIDIA GPU Cloud Machine Image, configuring settings as needed, then launching the VM. After launching the VM, you can SSH into it and start running deep learning jobs using framework containers from the NGC Container Registry.

This document provides step-by-step instructions for accomplishing this, including how to use the Azure CLI.
Chapter 2. BEFORE YOU START

Be sure you are familiar with the information in this chapter before starting to use the NVIDIA GPU Cloud Machine Image on Microsoft Azure.

2.1. Prerequisites

These instructions assume the following:

- You have an Azure account - https://portal.azure.com, with either permissions to create a Resource Group or with a Resource Group already available to you.
- Browsed the NGC website and identified an available NGC container and tag to run on the VMI.
- If you plan to use the Azure CLI or Terraform, then the Azure CLI 2.0 must be installed.
- You have SSH keys to use with Azure; see setup instructions below.
- Windows Users: The CLI code snippets are for bash on Linux or Mac OS X. If you are using Windows and want to use the snippets as-is, you can use the Windows Subsystem for Linux and use the bash shell (you will be in Ubuntu Linux).

Some of the CLI snippets in these instructions make use of jq, which should be installed on the machine from which you’ll run the CLI. You may paste these snippets into your own bash scripts or type them at the command line.

Additionally, if you plan to access locked NGC containers, you will need to perform the following steps from the NGC website (see NGC Getting Started Guide)

- Signed up for an NGC account at https://ngc.nvidia.com/signup.
- Created an NGC API key for access to locked containers within the NGC container registry.
2.2. Setting Up SSH Keys

If you do not already have SSH keys set up specifically for Azure, you will need to set one up and have it on the machine you will use to SSH to the VM. In the examples, the key is named "azure-key".

On Linux or OS X, generate a new key with the following command:

```
ssh-keygen -t rsa -b 2048 -f ~/.ssh/azure-key
```

On Windows, the location will depend on the SSH client you use, so modify the path above in the snippets or in your SSH client configuration.

2.3. Setting Up a Security Group

When creating your NVIDIA GPU Cloud VM, Azure sets up a network security group for the VM and you should choose to allow external access to inbound ports 22 (for SSH) and 443 (for HTTPS). You can add inbound rules to the network security group later for other ports as needed, such as port 8888 for DIGITS.

You can also set up a separate network security group so that it will be available any time you create a new NVIDIA GPU Cloud VM. This can be done ahead of time. Refer to the Microsoft instructions to Create, Change, or Delete a Network Security Group.

Add the following inbound rules to your network security group:

- **SSH**
  - Destination port ranges: 22
  - Protocol: TCP
  - Name: SSH

- **HTTPS**
  - Destination port ranges: 443
  - Protocol: TCP
  - Name: HTTPS

- **Others as needed**
  
  Example: DIGITS
  
  - Destination port ranges: 8888
  - Protocol: TCP
  - Name: DIGITS
Chapter 3.
LAUNCHING AN NVIDIA GPU CLOUD VM FROM THE AZURE PORTAL

3.1. Creating Your GPU Cloud VM

2. Select Create a Resource from the left-side menu.

3. On the New pane, search for "nvidia", and then select the NVIDIA GPU Cloud image that you want to use from the list - either the basic NVIDIA GPU Cloud image, or the NVIDIA GPU Cloud image with TensorFlow or PyTorch.
4. Click **Create** from the **NVIDIA GPU Cloud image** pane.

5. Complete the Basics settings as follows:
   - **Name**: Name of your choosing
   - **VM disk type**: SSD
   - **Username**: Your username
   - **Authentication type**: SSH public key
   - **SSH public key**: Paste in your SSH public key
   - **Resource group**: Use existing, then select your Resource group
6. Click OK.

7. From the **Choose a size** pane, filter the list by GPU compute type and SSD disk type, then select a P40, P100, or V100 SKU that meets your requirements and budget and click **Select**.

8. Under **Settings**, set up a Basic or Advanced Network Security Group.
   - To use the network security group assigned to this VM, select **Basic** and then select ports SSH (22) and HTTPS (443). You can add inbound rules to the network security group later for other ports as needed.
   - To use an existing network security group, select **Advanced**, then click the Network security group (firewall) arrow, and then select one of the already created NSGs.
You can also create a new NSG at this time by clicking Create new and then adding inbound rules.

See the section Setting Up a Network Security Group for the list of rules to add.

9. Make other Settings selections as needed, then click OK.

After the validation passes, the portal presents the details of your new image which you can download as a template to automate deployment later.

10. Click OK to deploy the image.

The deployment starts, as indicated by the traveling bar underneath the Alert icon.

It may take a few minutes to complete. A pop-up alert will let you know when the VM instance is successfully deployed.
3.2. Connecting to Your GPU Cloud VM Instance

1. Open the page for your GPU Cloud VM instance.
   a) Select Virtual machines from the left side menu.
   b) Locate and select your GPU Cloud VM.

2. Click Connect from the top menu, then follow the instructions to establish an SSH connection to your VMI.

   If you plan to access locked NGC containers, you will need to log in to the NGC container registry. See Logging in to the NGC Container Registry) for instructions.

   If the instructions for SSH login do not work, see the Troubleshooting SSH connections to an Azure Linux VM that fails, errors out, or is refused documentation for additional information.

3.3. Starting and Stopping Your GPU Cloud VM

1. Open the page for your GPU Cloud VM.
   a) Select Virtual machines from the left side menu.
   b) Locate and select your GPU Cloud VM.

2. Use the menu at the top of your GPU Cloud VM instance page to Start, Restart, or Stop your VM.

3.4. Deleting Your GPU Cloud VM and Associated Resources
When you created your VM, other resources for that instance were automatically created for you, such as a network interface, public IP address, and boot disk. If you deleted your VM, you will also need to delete these resources.

1. Click **All resources** from the left side menu.
2. Click the check box for the GPU VM that you want to delete, as well as the associated disk and network interface.

The associated resources use the same base name as your VM and then appends an additional identifier.

3. Click **Delete** from the top menu.
4. Enter ‘yes’ in the confirmation box on the Delete Resources pane and then click **Delete**.
5. After you’ve deleted the VM disk and network interface, delete the public IP address resource.
   a) Click the Public IP address check box for your VM and then click Delete.
   b) Enter ‘yes’ in the confirmation box on the Delete Resources pane and then click Delete.
Chapter 4.
LAUNCHING AN NVIDIA GPU CLOUD VM USING AZURE CLI

This section explains how to create a GPU Cloud VM instance using the Azure CLI. For complete CLI documentation and sample scripts visit the Azure CLI 2.0 Documentation.

Using Example Python Scripts
A comprehensive set of example Python scripts for automating the CLI are provided at https://github.com/nvidia/ngc-examples/tree/master/ncsp. You can download the scripts and modify them to meet your requirements. The code examples that follow use similar environment variables and structure as the scripts.

Using the Instructions in this Chapter
This flow and the code snippets in this section are for Linux or Mac OS X. If you are using Windows, you can use the Windows Subsystem for Linux and use the bash shell (where you will be in Ubuntu Linux). Many of these CLI commands can have significant delays.

4.1. Installing Azure CLI
Follow the instructions at https://docs.microsoft.com/en-us/cli/azure/install-azure-cli. These include instructions for Linux, Mac, and Windows.

4.2. Preparing Your VM Variables
Use the following table as a guide for determining the values you will need for creating your GPU Cloud VM. The variable names are arbitrary, and used in the instructions that follow.

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>DESCRIPTION</th>
<th>EXAMPLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>AZ_VM_NAME</td>
<td>Name for your GPU Cloud VM</td>
<td>my-nvgpu-vmi</td>
</tr>
<tr>
<td>AZ_RESOURCE_GROUP</td>
<td>Your resource group</td>
<td>ACME_RG</td>
</tr>
<tr>
<td>AZ_IMAGE</td>
<td>The GPU Cloud VMI. See the release notes <a href="https://docs.nvidia.com/ngc/ngc-azure-vmi-release-notes">https://docs.nvidia.com/ngc/ngc-azure-vmi-release-notes</a> for the latest release.</td>
<td>NVIDIA-GPU-Cloud-Image</td>
</tr>
<tr>
<td>AZ_LOCATION</td>
<td>A zone that contains GPUs. Refer to <a href="https://azure.microsoft.com/en-us/global-infrastructure/services/">https://azure.microsoft.com/en-us/global-infrastructure/services/</a> to see available locations for NCv2 and NCv3 series SKUs.</td>
<td>westus2</td>
</tr>
<tr>
<td>AZ_SIZE</td>
<td>The SKU specified by the number of vCPUs, RAM, and GPUs. Refer to <a href="https://docs.microsoft.com/en-us/azure/virtual-machines/linux/sizes-gpu">https://docs.microsoft.com/en-us/azure/virtual-machines/linux/sizes-gpu</a> for the list of P40, P100, and V100 SKUs to choose from.</td>
<td>NC6s_v2</td>
</tr>
<tr>
<td>AZ_SSH_KEY</td>
<td><code>&lt;path&gt;/public-azure-key.pub</code></td>
<td>~/.ssh/azure-key.pub</td>
</tr>
<tr>
<td>AZ_USER</td>
<td>Your username</td>
<td>jsmith</td>
</tr>
<tr>
<td>AZ_NSG</td>
<td>Your network security group</td>
<td>my-nvgpu-nsg</td>
</tr>
</tbody>
</table>

4.3. Creating Your GPU Cloud VM

Be sure you have installed Azure CLI and that you are ready with the VM setup information listed in the section Preparing Your VM Variables. You can then either
manually replace the variable names in the commands in this section with the actual values, or define the variables ahead of time.

1. Log in to the Azure CLI.

   `az login`

2. Enter the following:

   ```
   az vm create \
   --name ${AZ_VM_NAME} \
   --resource-group ${AZ_RESOURCE_GROUP} \
   --image ${AZ_IMAGE} \ 
   --location ${AZ_LOCATION} \ 
   --size ${AZ_SIZE} \ 
   --admin-username ${AZ_USER} \ 
   --nsg ${AZ_NSG} 
   ```

   If successful, you should see output consisting of a JSON description of your VM. The GPU Cloud VM gets deployed.

   Note the public IP address for use when establishing an SSH connection to the VM. You can also set up an AZ_PUBLIC_IP variable by defining an Azure JSON file for the VM as follows:

   ```
   AZ_JSON=$(az vm create \
   --name ${AZURE_VM_NAME} \
   --resource-group ${AZ_RESOURCE_GROUP} \
   --image ${AZ_IMAGE} \ 
   --location ${AZ_LOCATION} \ 
   --size ${AZ_SIZE} \ 
   --admin-username ${AZ_USER} \ 
   --nsg ${AZ_NSG})
   AZ_PUBLIC_IP=$(echo $AZ_JSON | jq .publicIpAddress | sed 's/"//g') && \
   echo $AZ_JSON && echo AZ_PUBLIC_IP=$AZ_PUBLIC_IP
   ```

Azure sets up a non-persistent scratch disk for each VM. See the sections Using Premium Storage SSD for Datasets and Using File Storage for Datasets for instructions on setting up alternate storage for your datasets.

**4.4. Connecting to Your GPU Instance with SSH**

Run `ssh` to connect to your GPU VM.instance.

```
ssh -i $AZ_SSH_KEY $AZ_USER@$AZ_PUBLIC_IP
```

**4.5. Stopping (Deallocating) and Starting VMs with the CLI**

VMs can be stopped and started again without losing any of their storage and other resources.

To stop and deallocate a running VM:

```
az vm deallocate --resource-group $AZ_RESOURCE_GROUP --name $AZ_VM_NAME
```
To start a stopped VM:

```bash
az vm start --resource-group $AZ_RESOURCE_GROUP --name $AZ_VM_NAME
```

When starting a stopped VM, you will need to update the public IP variable, as it will change with the newly started VM.

```bash
AZ_PUBLIC_IP=$(az network public-ip show \
   --resource-group $AZ_RESOURCE_GROUP \
   --name $AZ_VM_NAME\PublicIP | jq .ipAddress | sed 's/"//g') && \
echo AZ_PUBLIC_IP=$AZ_PUBLIC_IP
```

### 4.6. Deleting VMs and Associated Resources with the CLI

When you created your VM, other resources for that instance were automatically created for you, such as a network interface, public IP address, and boot disk. If you deleted your instance, you will also need to delete these resources.

Perform the deletions in the following order.

1. Delete your VM.

   ```bash
   az vm delete -g $AZ_RESOURCE_GROUP -n $AZ_VM_NAME
   ```

2. Delete the VM OS disk.
   a) List the disks in your Resource Group.

   ```bash
   az disk list -g $AZ_RESOURCE_GROUP
   ```
   The associated OS disk will have the name of your VM as the base name.

   b) Delete the OS disk.

   ```bash
   az disk delete -g $AZ_RESOURCE_GROUP -n MyDisk
   ```

3. Delete the VM network interface.
   a) List the network interface resources in your Resource Group.

   ```bash
   az network nic list -g $AZ_RESOURCE_GROUP
   ```
   The associated network interface will have the name of your VM as the base name.

   b) Delete the network interface resource.

   ```bash
   az network nic delete -g $AZ_RESOURCE_GROUP -n MyNic
   ```

4. Delete the VM public IP address.
   a) List the public IPs in your Resource Group.

   ```bash
   az network public-ip list -g $AZ_RESOURCE_GROUP
   ```
The associated public IP will have the name of your VM as the base name.
b) Delete the public IP.

```
az network public-ip delete -g $AZ_RESOURCE_GROUP -n MyIp
```

Chapter 5.
USING PREMIUM STORAGE SSDS FOR DATASETS

You can create Premium Storage SSD from the Azure dashboard. Premium Storage SSDs are ideal for persistent storage of a large number of datasets and offer better performance.

Unlike File Storage, Premium Storage SSDs cannot be shared across multiple VMs. To share persistent data storage, you need to use File Storage. See the section Using File Storage for Datasets.

5.1. Creating Data Disk Storage Using the Azure Dashboard

1. Open the Azure Dashboard, select Virtual machines from the left side menu, then select your VM.
2. In the Settings section, click Disks, then click +Add data disk.
3. Click the NAME list arrow, then click **Create disk** at the top of the list.

4. At the Create Managed Disk pane,
   - Enter a disk name
   - Select a resource group
   - Select Premium SSD for Account type
   - Enter a disk size

5. Click **Create**.

6. When the validation is completed, click **Save**.

5.2. Creating Data Disk Storage Using the Azure CLI

To create a new data disk and attach it to your VM, include the following option in the `az vm create` command.

```
--data-disk-sizes-gb <data-disk-size>
```

To attach an existing data disk to your VM when creating it, include the following option in the `az vm create` command.

```
--attach-data-disks <data-disk-name>
```
5.3. Mounting a Data Disk

1. Once the data disk is created, establish an SSH connection to your VM.

2. Create a filesystem on the data disk.

   You can view the volume by running lsblk command.

   ```
   ~# lsblk
   NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT
   sdb 8:16 0 1.5T 0 disk
   └─sdb1 8:17 0 1.4T 0 part /mnt
   sr0 11:0 1 628K 0 rom
   sdc 8:32 0 2T 0 disk
   └─sdc1 8:33 0 2T 0 part
   sda 8:0 0 240G 0 disk
   └─sda1 8:1 0 240G 0 part /
   :
   ~# mkfs.ext4 /dev/sdc1
   ```

3. Mount the volume to a mount directory.

   ```
   ~# mount /dev/sdc1 /data
   ```

   To mount the volume automatically every time the VM is stopped and restarted, add an entry to `/etc/fstab`.

   When adding an entry to `/etc/fstab`, use a UUID based device path (See device-names-problem for details).

   For example:

   ```
   UUID=33333333-3b3b-3c3c-3d3d-3e3e3e3e3e3e /data ext4 defaults,nofail 1 2
   ```

5.4. Deleting a Data Disk

You can delete a Data Disk only if it not attached to a VM. Be aware that once you delete a Data Disk, you cannot undelete it.

1. Open the Azure Dashboard and click All resources from the left side menu.

2. Filter by Disks type, then locate and select the check box for your data disk.

3. Click Delete.

4. Enter ‘yes’ to confirm, then click Delete.
Delete Resources
Deleting 1 resource

Do you want to delete all the selected resources?

Warning! Deleting the selected resources is irreversible. This will permanently delete the selected resources, their related resources and contents. If you are not sure about the selected resource dependencies, please go to individual resource type blade to perform the delete operation. This action cannot be undone. Do you want to continue?

Confirm delete:

- [ ] Yes

Selected resources:

- [ ] cce-test-datadisk2 (Disk)

Delete  Cancel
Chapter 6.
USING FILE STORAGE FOR DATASETS

You can create persistent file storage from the Azure Dashboard. File storage is useful for sharing datasets across multiple VMs.

For working with a large number of small files, such as in a dataset, Premium Storage SSDs offer better performance. For instructions on setting up and using Premium Storage SSDs, see the section Using Premium Storage SSD for Datasets.

6.1. Creating a Storage Account

You need a storage account to set up your file storage.

1. Log in to the Azure portal.
2. Click All services from the left side menu, then click Storage accounts (under the STORAGE section).
3. Click Add.
4. Enter information as follows:
   - Name: enter any name.
   - Deployment model: Resource Manager
   - Account kind: Select based on your storage needs and cost.
Select the subscription in which you want to create the new storage account.
- **Resource group**: Use existing
- **Select the geographic region.**

5. Click **Create** to create the storage account.

When you create a storage account, Azure generates two 512-bit storage access keys, which are used for authentication when the storage account is accessed.

### 6.2. Creating File Storage

1. Click **Storage accounts** from the left side menu, then click your storage account.
2. Under Services, click **Files**.
3. Click **+File share**, then enter a name for your storage and required quota and click **OK**.
4. Note the keys and use it to mount.
5. Ensure that the region selected above is same as the location of your GPU Cloud VM.

### 6.3. Mounting the File Storage

1. Once file storage created, establish an SSH connection to your VM.
2. Mount the share to the local directory (pass your StorageAccount and key as passwords).

```bash
sudo mount -t cifs \\
//myCifsAccount.file.core.windows.net/cifsShare01 /cifs \\
-o vers=3.0,username=myCifsAccount,password=mystorageaccountkey,\ 
dir_mode=0777,file_mode=0777
```

or, for higher performance, use the following.

```bash
sudo mount -t cifs \\
//myCifsAccount.file.core.windows.net/cifsShare01 /cifs \\
-o vers=2.1,username=myCifsAccount,password=mystorageaccountkey,\ 
dir_mode=0777,file_mode=0777,cache=strict
```

3. **[Optional]** To persist the mount through reboots, an entry to **/etc/fstab**.

```bash
//myCifsAccount.file.core.windows.net/cifsShare01 /mymountpoint cifs vers=3.0,username=myCifsAccount,password=mystorageaccountkey==,\ 
dir_mode=0777,file_mode=0777
```

or, for higher performance, use the following.

```bash
//myCifsAccount.file.core.windows.net/cifsShare01 /mymountpoint cifs vers=2.1,username=myCifsAccount,password=mystorageaccountkey==,\ 
dir_mode=0777,file_mode=0777,cache=strict
```

4. Copy the dataset over.

```bash
scp -r local_dataset_dir/ <username>@<Azure_VM_Instance>:/data
```

The Storage account and mount should be in the same region.
6.4. Deleting File Storage

Be aware that once you delete your File Storage, you cannot undelete it.

1. Open the Azure Dashboard and select **Storage accounts** from the left side menu.
2. Locate and click your storage account.
3. Click **Files** from the Services section.

4. From the list, click the file share to delete.

5. Click **Delete share** from the top menu.
6. Follow the instructions in the **Delete file share** pane and then click **OK**.
Chapter 7. EXAMPLES OF RUNNING CONTAINERS

This chapter walks you through the process of logging in to the NGC container registry, pulling and running a container, and using file storage and data disks for storage.

7.1. Logging Into the NGC Container Registry

You need to log in to the NGC container registry only if you want to access locked containers from the registry. Most of the NGC containers are freely available (unlocked) and do not require an NGC account or NGC API key.

You do not need to log into the NGC container registry if you are using either the NVIDIA GPU Cloud image with PyTorch or the NVIDIA GPU Cloud image with TensorFlow and intend to use the containers already built into the image.

If necessary, log in to the NGC container registry manually by running the following script from the VMI.

```
ngc-login.sh <your-NGC-API-key>
```

From this point you can run Docker commands and access locked NGC containers from the VM instance.

7.2. Running a Container

This section explains the basic process for running a container on the NVIDIA GPU Cloud image for TensorFlow, the NVIDIA GPU Cloud image for PyTorch, and the basic NVIDIA GPU Cloud image.
Running the Built-in TensorFlow Container

To run the TensorFlow container in the VM created from the NVIDIA GPU Cloud image for TensorFlow, refer to the release notes for the correct tag to use, then enter the following command.

```bash
docker run --runtime=nvidia --rm -it nvcr.io/nvidia/tensorflow:<tag>
```

Running the Built-in PyTorch Container

To run the PyTorch container in the VM created from the NVIDIA GPU Cloud image, refer to the release notes for the correct tag to use, then enter the following command.

```bash
docker run --runtime=nvidia --rm -it nvcr.io/nvidia/pytorch:<tag>
```

Running a Container from the NGC Container Registry

To run containers from the NGC container registry,

1. If necessary, log in to the NGC container registry as explained in the previous section.
2. Enter the following commands.

```bash
docker pull nvcr.io/nvidia/<container-image>:<tag>
docker run --runtime=nvidia --rm -it nvcr.io/nvidia/<container-image>:<tag>
```

7.3. Example: MNIST Training Run Using TensorFlow Container

Once logged in to the NVIDIA GPU Cloud VM, you can run the MNIST example under TensorFlow.

Note that the TensorFlow built-in example will pull the MNIST dataset from the web.

1. Pull and run the TensorFlow container.

```bash
docker pull nvcr.io/nvidia/tensorflow:18.08-py3
docker run --runtime=nvidia --rm -it nvcr.io/nvidia/tensorflow:18.08-py3
```
2. Following this tutorial: https://www.tensorflow.org/get_started/mnist/beginners, run the `MNIST_with_summaries` example.

```bash
cd /opt/tensorflow/tensorflow/examples/tutorials/mnist
python mnist_with_summaries.py
```