# ubuntu18.04安装pytorch

# 1.CPU版本Pytorch安装

CPU版本的pytoch安装比较简单，在Pytorch官网中选择python版本，运行conda或者pip命令即可，如下图所示。注意，为了方便多个Python版本的管理，通常需要在anaconda中创建新的环境空间。

# 2.GPU版本Pytorch安装

GPU版本的pytorch安装比较复杂，在安装pytorch之前，通常需要安装显卡驱动，cuda和cudnn，**CUDA**是NVIDIA推出的用于自家GPU的**并行计算框架**，也就是说CUDA只能在NVIDIA的GPU上运行；**cuDNN**（CUDA Deep Neural Network library）：是NVIDIA打造的针对深度神经网络的**加速库**，是一个用于深层神经网络的GPU加速库。如果你要用GPU训练模型，**cuDNN不是必须的，但是一般会采用这个加速库**。CUDA是必须的，cudnn是可选的。

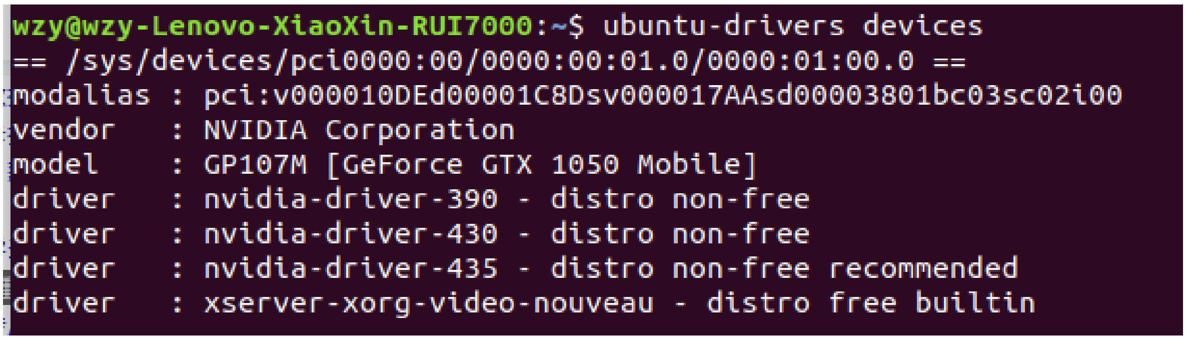
通常需要考虑：

**（1）检查显卡型号是否支持GPU运算**

检查方法：终端输入

```
ubuntu-drivers devices
```

得到以下结果：

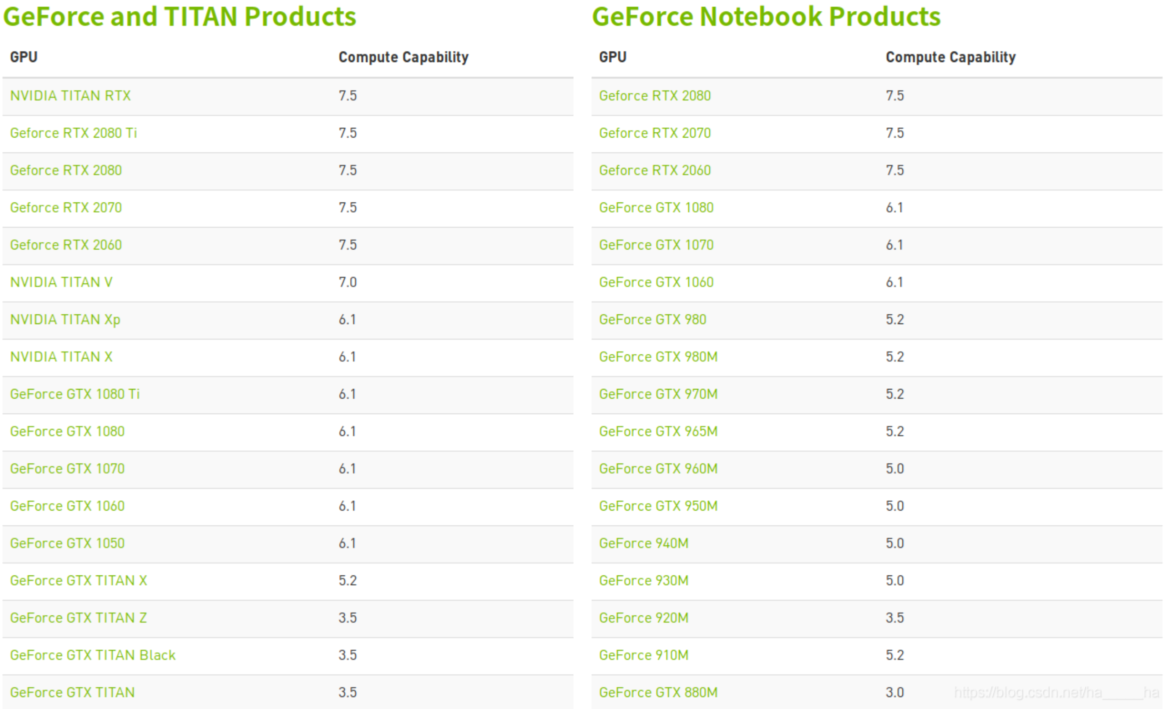


```
wzy@wzy-Lenovo-XiaoXin-RUI7000:~$ ubuntu-drivers devices
== /sys/devices/pci0000:00/0000:00:01.0/0000:01:00.0 ==
modalias : pci:v000010DEd00001C8Dsv000017AAsd00003801bc03sc02i00
vendor   : NVIDIA Corporation
model    : GP107M [GeForce GTX 1050 Mobile]
driver   : nvidia-driver-390 - distro non-free
driver   : nvidia-driver-430 - distro non-free
driver   : nvidia-driver-435 - distro non-free recommended
driver   : xserver-xorg-video-nouveau - distro free builtin
```

图中，“model”对应的就是电脑显卡型号，“driver”中后面带有“recommanded”就是推荐的显卡驱动型号。根据显卡型号到英伟达官网查询是否支持GPU。

网站显示如下：

先选择“CUDA GeForce and TITAN Products”，再看具体型号，电脑GPU为GTX 1050，支持GPU 运算。



### GeForce and TITAN Products

| GPU | Compute Capability |
|---|---|
| NVIDIA TITAN RTX | 7.5 |
| Geforce RTX 2080 Ti | 7.5 |
| Geforce RTX 2080 | 7.5 |
| Geforce RTX 2070 | 7.5 |
| Geforce RTX 2060 | 7.5 |
| NVIDIA TITAN V | 7.0 |
| NVIDIA TITAN Xp | 6.1 |
| NVIDIA TITAN X | 6.1 |
| GeForce GTX 1080 Ti | 6.1 |
| GeForce GTX 1080 | 6.1 |
| GeForce GTX 1070 | 6.1 |
| GeForce GTX 1060 | 6.1 |
| GeForce GTX 1050 | 6.1 |
| GeForce GTX TITAN X | 5.2 |
| GeForce GTX TITAN Z | 3.5 |
| GeForce GTX TITAN Black | 3.5 |
| GeForce GTX TITAN | 3.5 |

### GeForce Notebook Products

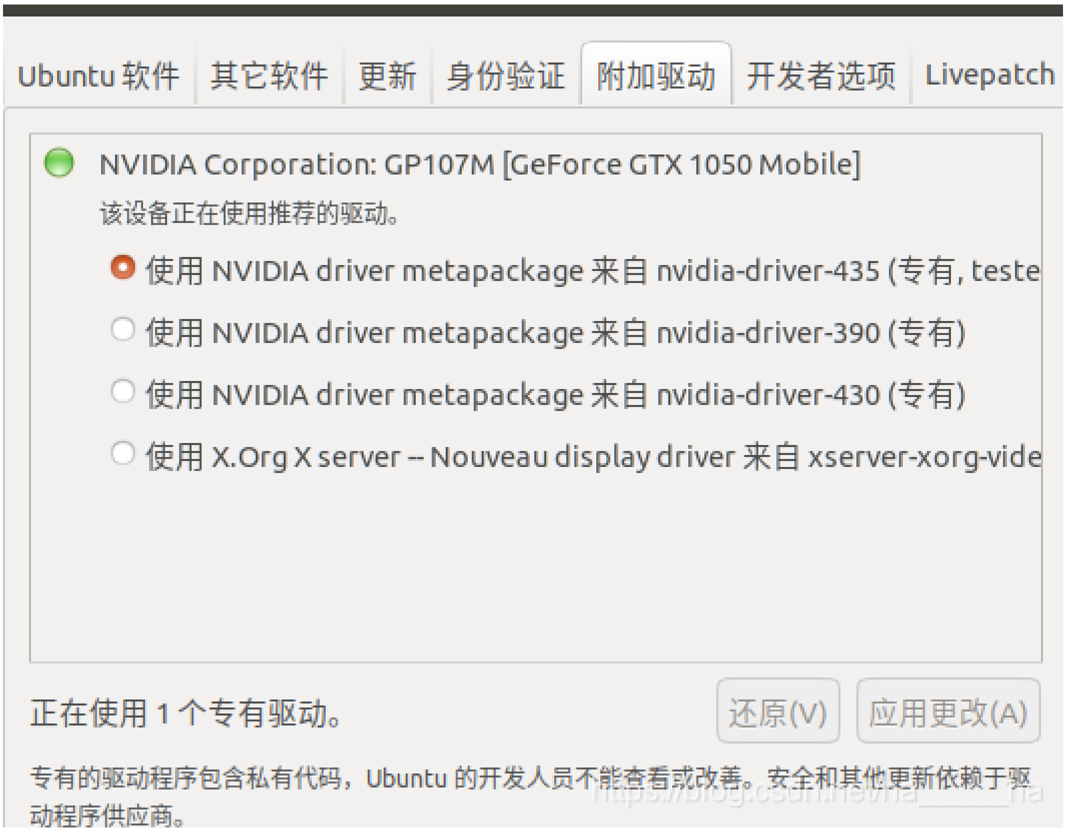
| GPU | Compute Capability |
|---|---|
| Geforce RTX 2080 | 7.5 |
| Geforce RTX 2070 | 7.5 |
| Geforce RTX 2060 | 7.5 |
| GeForce GTX 1080 | 6.1 |
| GeForce GTX 1070 | 6.1 |
| GeForce GTX 1060 | 6.1 |
| GeForce GTX 980 | 5.2 |
| GeForce GTX 980M | 5.2 |
| GeForce GTX 970M | 5.2 |
| GeForce GTX 965M | 5.2 |
| GeForce GTX 960M | 5.0 |
| GeForce GTX 950M | 5.0 |
| GeForce 940M | 5.0 |
| GeForce 930M | 5.0 |
| GeForce 920M | 3.5 |
| GeForce 910M | 5.2 |
| GeForce GTX 880M | 3.0 |

**（2）安装显卡驱动**

确定显卡支持GPU 运算后，接下来需要安装显卡驱动，目前尝试过两种安装方法，都成功了。

- 安装方法一：

系统“软件和更新”，“附加驱动”，选择推荐的驱动安装。



NVIDIA Corporation: GP107M [GeForce GTX 1050 Mobile]

该设备正在使用推荐的驱动。

● 使用 NVIDIA driver metapackage 来自 nvidia-driver-435 (专有, teste

○ 使用 NVIDIA driver metapackage 来自 nvidia-driver-390 (专有)

○ 使用 NVIDIA driver metapackage 来自 nvidia-driver-430 (专有)

○ 使用 X.Org X server -- Nouveau display driver 来自 xserver-xorg-vide

正在使用 1 个专有驱动。

专有的驱动程序包含私有代码，Ubuntu 的开发人员不能查看或改善。安全和其他更新依赖于驱动程序供应商。

- 安装方法二：

运行以下命令，安装驱动

```
sudo apt install nvidia-435
```

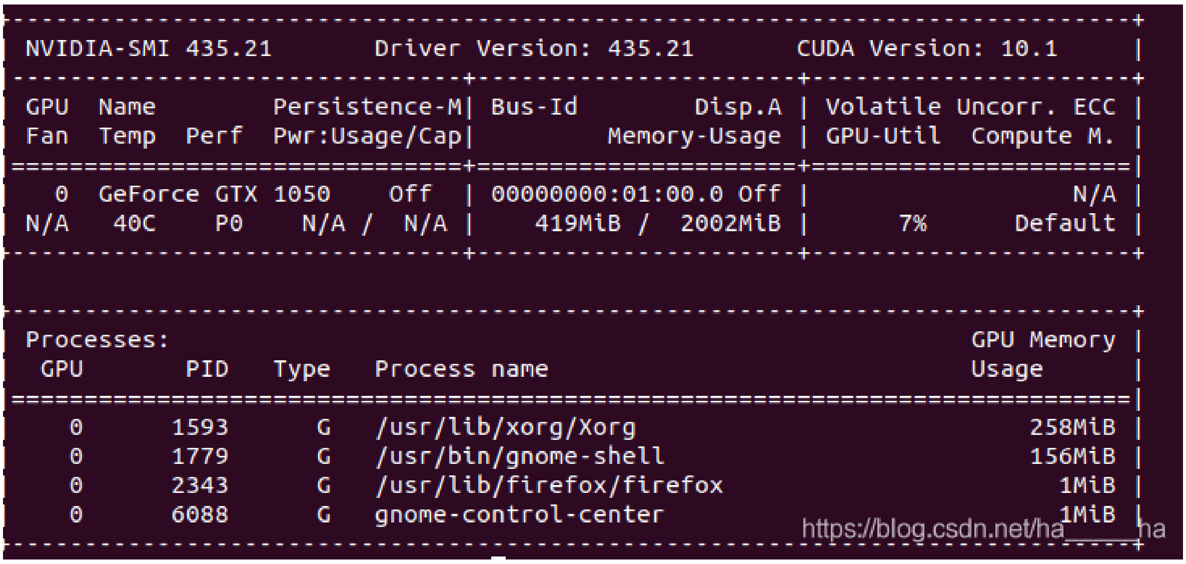
**判断驱动是否安装成功**

方法一：

终端输入命令

```
sudo nvidia-smi
```

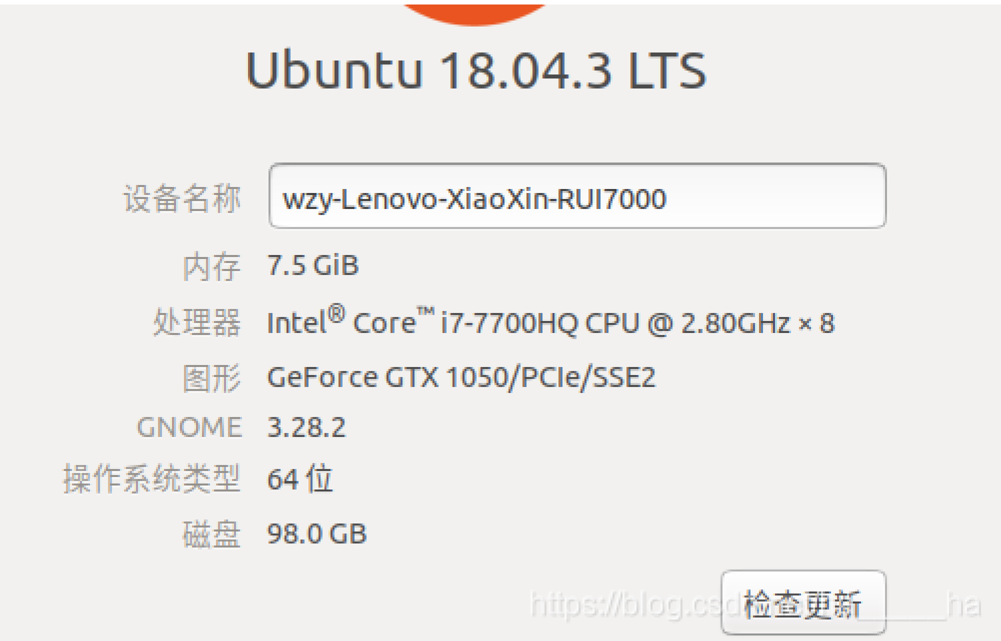
显示以下信息则表示安装成功



```
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 435.21       Driver Version: 435.21       CUDA Version: 10.1     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  GeForce GTX 1050    Off  | 00000000:01:00.0 Off |                  N/A |
| N/A   40C    P0    N/A /  N/A |    419MiB /  2002MiB |      7%      Default |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================|
|    0      1593      G   /usr/lib/xorg/Xorg                           258MiB |
|    0      1779      G   /usr/bin/gnome-shell                         156MiB |
|    0      2343      G   /usr/lib/firefox/firefox                       1MiB |
|    0      6088      G   gnome-control-center                           1MiB |
+-----------------------------------------------------------------------------+
```

方法二：

系统设置查看“图形”信息，变为英伟达的显卡类型即可。



Ubuntu 18.04.3 LTS

设备名称 wzy-Lenovo-XiaoXin-RUI7000

内存 7.5 GiB

处理器 Intel® Core™ i7-7700HQ CPU @ 2.80GHz × 8

图形 GeForce GTX 1050/PCIe/SSE2

GNOME 3.28.2

操作系统类型 64 位

磁盘 98.0 GB

（4）安装cuda

cuda和cudnn、pytorch版本之间要保证对应，能够支持。以下网址中有之前各版本的cuda下载（[https://developer.nvidia.com/cuda-toolkit-archive](https://developer.nvidia.com/cuda-toolkit-archive)），本文选择的是cuda10.2，根据系统环境选择合适的版本下载，有补丁的连补丁一同下载。关于Ubuntu系统一项，记住一个原则：高版本兼容低版本。

## 查看显卡CUDA支持情况：

[https://developer.nvidia.com/cuda-gpus](https://developer.nvidia.com/cuda-gpus)

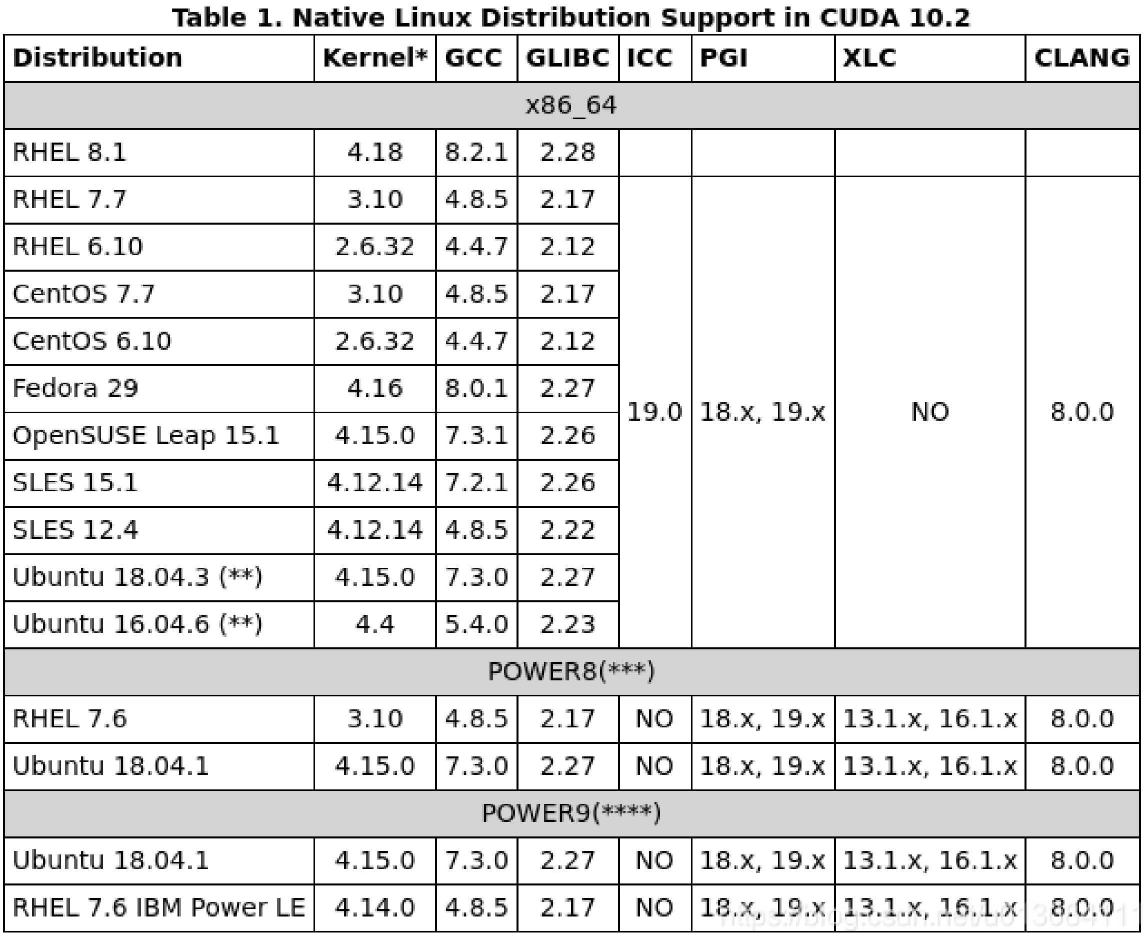


**Table 1. Native Linux Distribution Support in CUDA 10.2**

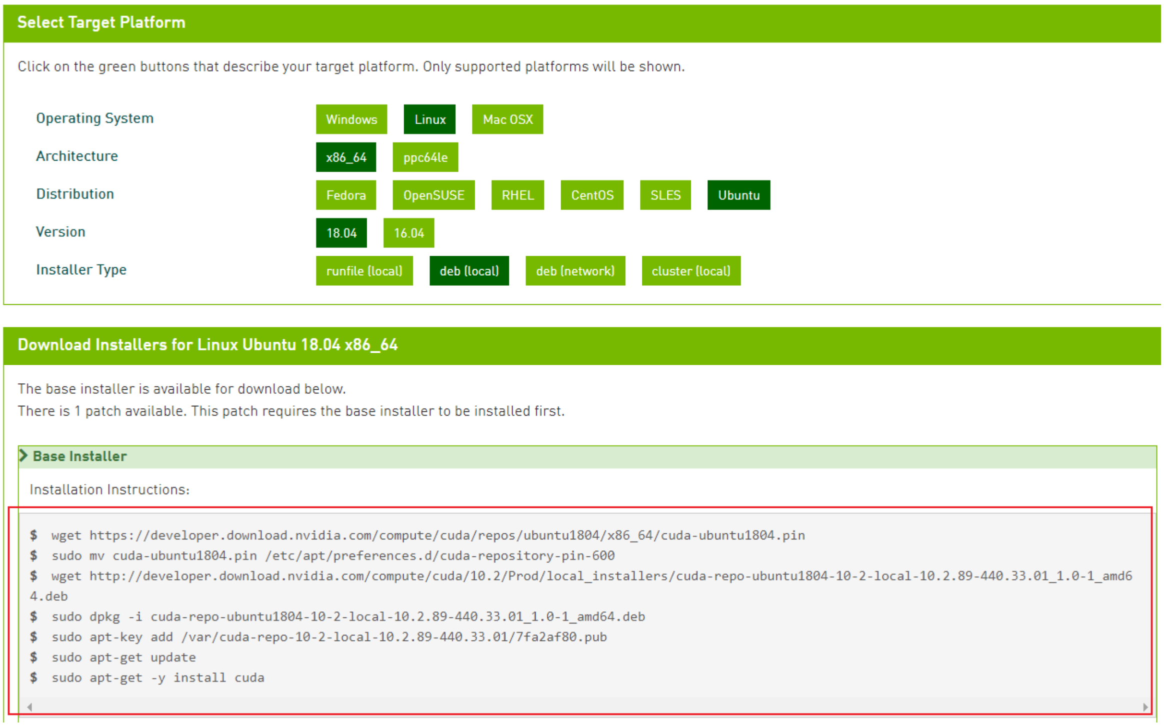
| Distribution | Kernel* | GCC | GLIBC | ICC | PGI | XLC | CLANG |
|---|---|---|---|---|---|---|---|
| x86_64 | | | | | | | |
| RHEL 8.1 | 4.18 | 8.2.1 | 2.28 | | | | |
| RHEL 7.7 | 3.10 | 4.8.5 | 2.17 | 19.0 | 18.x, 19.x | NO | 8.0.0 |
| RHEL 6.10 | 2.6.32 | 4.4.7 | 2.12 | | | | |
| CentOS 7.7 | 3.10 | 4.8.5 | 2.17 | | | | |
| CentOS 6.10 | 2.6.32 | 4.4.7 | 2.12 | | | | |
| Fedora 29 | 4.16 | 8.0.1 | 2.27 | | | | |
| OpenSUSE Leap 15.1 | 4.15.0 | 7.3.1 | 2.26 | | | | |
| SLES 15.1 | 4.12.14 | 7.2.1 | 2.26 | | | | |
| SLES 12.4 | 4.12.14 | 4.8.5 | 2.22 | | | | |
| Ubuntu 18.04.3 (**) | 4.15.0 | 7.3.0 | 2.27 | | | | |
| Ubuntu 16.04.6 (**) | 4.4 | 5.4.0 | 2.23 | | | | |
| POWER8(***) | | | | | | | |
| RHEL 7.6 | 3.10 | 4.8.5 | 2.17 | NO | 18.x, 19.x | 13.1.x, 16.1.x | 8.0.0 |
| Ubuntu 18.04.1 | 4.15.0 | 7.3.0 | 2.27 | NO | 18.x, 19.x | 13.1.x, 16.1.x | 8.0.0 |
| POWER9(****) | | | | | | | |
| Ubuntu 18.04.1 | 4.15.0 | 7.3.0 | 2.27 | NO | 18.x, 19.x | 13.1.x, 16.1.x | 8.0.0 |
| RHEL 7.6 IBM Power LE | 4.14.0 | 4.8.5 | 2.17 | NO | 18.x, 19.x | 13.1.x, 16.1.x | 8.0.0 |

## 查看本机显卡

```
$ lspci | grep -i nvidia
```

# 安装CUDA

根据驱动版本选择CUDA版本：
[https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html](https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html)
CUDA10.2下载：
[https://developer.nvidia.com/cuda-downloads?target_os=Linux&target_arch=x86_64&target_distro=Ubuntu&target_version=1804&target_type=deblocal](https://developer.nvidia.com/cuda-downloads?target_os=Linux&target_arch=x86_64&target_distro=Ubuntu&target_version=1804&target_type=deblocal)



Select Target Platform

Click on the green buttons that describe your target platform. Only supported platforms will be shown.

Operating System: Windows [✓ Linux] Mac OSX

Architecture: [✓ x86_64] ppc64le

Distribution: Fedora OpenSUSE RHEL CentOS SLES [✓ Ubuntu]

Version: [✓ 18.04] 16.04

Installer Type: runfile (local) [✓ deb (local)] deb (network) cluster (local)

Download Installers for Linux Ubuntu 18.04 x86_64

The base installer is available for download below.
There is 1 patch available. This patch requires the base installer to be installed first.

Base Installer

Installation Instructions:

```
$  wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64/cuda-ubuntu1804.pin
$  sudo mv cuda-ubuntu1804.pin /etc/apt/preferences.d/cuda-repository-pin-600
$  wget http://developer.download.nvidia.com/compute/cuda/10.2/Prod/local_installers/cuda-repo-ubuntu1804-10-2-local-10.2.89-440.33.01_1.0-1_amd6
4.deb
$  sudo dpkg -i cuda-repo-ubuntu1804-10-2-local-10.2.89-440.33.01_1.0-1_amd64.deb
$  sudo apt-key add /var/cuda-repo-10-2-local-10.2.89-440.33.01/7fa2af80.pub
$  sudo apt-get update
$  sudo apt-get -y install cuda
```

CUDA安装命令：

```
$ wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64/cuda-ubuntu1804.pin
$ sudo mv cuda-ubuntu1804.pin /etc/apt/preferences.d/cuda-repository-pin-600
$ wget http://developer.download.nvidia.com/compute/cuda/10.2/Prod/local_installers/cuda-repo-ubuntu1804-10-2-local-10.2.89-440.33.01_1.0-1_amd64.deb
$ sudo dpkg -i cuda-repo-ubuntu1804-10-2-local-10.2.89-440.33.01_1.0-1_amd64.deb
$ sudo apt-key add /var/cuda-repo-10-2-local-10.2.89-440.33.01/7fa2af80.pub
$ sudo apt-get update
$ sudo apt-get -y install cuda
```

# 安装后

添加环境变量

```
$ sudo vim ~/.bashrc
```

在最后添加

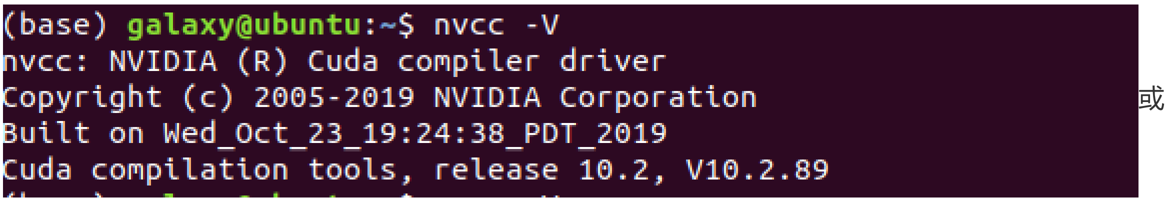
```
$ export PATH=/usr/local/cuda-10.2/bin:/usr/local/cuda-10.2/NsightCompute-2019.1${PATH:+:${PATH}}
$ export LD_LIBRARY_PATH=/usr/local/cuda-10.2/lib64\${LD_LIBRARY_PATH:+:${LD_LIBRARY_PATH}}
```

重新加载

```
$ source ~/.bashrc
```

查看CUDA是否成功安装

```
$ nvcc -V
```

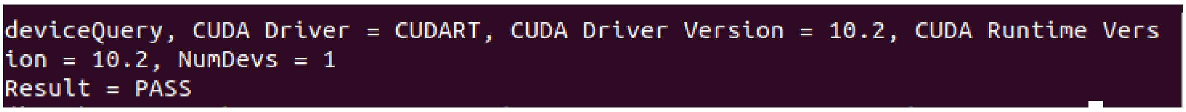


```
(base) galaxy@ubuntu:~$ nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2019 NVIDIA Corporation
Built on Wed_Oct_23_19:24:38_PDT_2019
Cuda compilation tools, release 10.2, V10.2.89
```

或者，输入以下命令验证

```
$ cd /usr/local/cuda-10.2/samples/1_Utilities/deviceQuery
$ sudo make
$ ./deviceQuery
```

结果的最后几行为



```
deviceQuery, CUDA Driver = CUDART, CUDA Driver Version = 10.2, CUDA Runtime Vers
ion = 10.2, NumDevs = 1
Result = PASS
```

# 安装cuDNN

注册并下载以下3个文件
cuDNN Runtime Library for Ubuntu18.04 (Deb)
cuDNN Developer Library for Ubuntu18.04 (Deb)
cuDNN Code Samples and User Guide for Ubuntu18.04 (Deb)

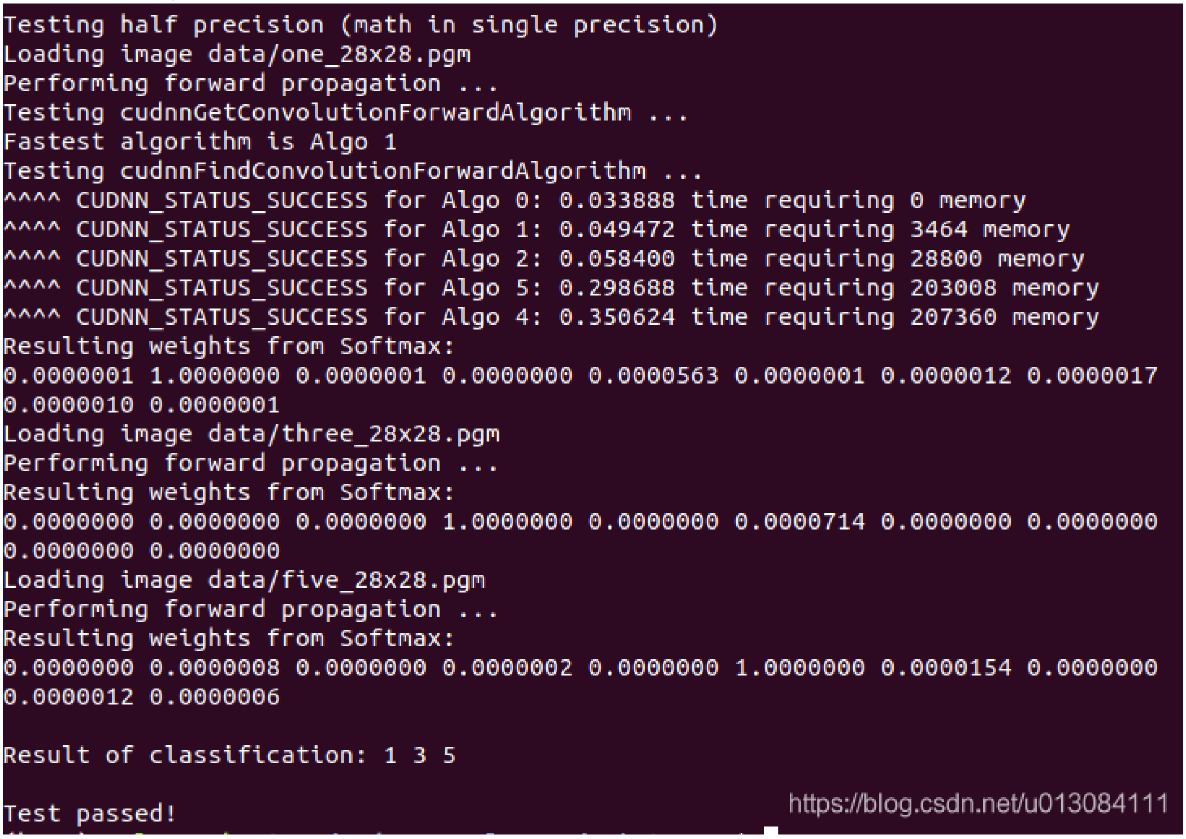
** 要下载deb文件，不要下载tgz文件！ （亲测tgz易错）
网址
https://developer.nvidia.com/rdp/cudnn-download
进入到下载文件所在目录

```
$ sudo dpkg -i libcudnn7_7.6.5.32-1+cuda10.2_amd64.deb
$ sudo dpkg -i libcudnn7-dev_7.6.5.32-1+cuda10.2_amd64.deb
$ sudo dpkg -i libcudnn7-doc_7.6.5.32-1+cuda10.2_amd64.deb
```

检测cuDNN是否安装成功：
运行/usr/src/cudnn_samples_v7 中的mnistCUDNN sample

```
$ cd /usr/src/cudnn_samples_v7
$ cp -r /usr/src/cudnn_samples_v7/ $HOME
$ cd  $HOME/cudnn_samples_v7/mnistCUDNN
$ make clean && make
$ ./mnistCUDNN
```

如果出现 Test passed! 则说明安装成功，结果如图

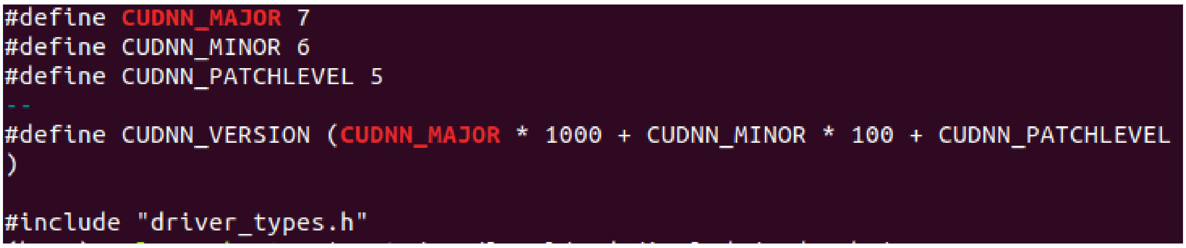


```
Testing half precision (math in single precision)
Loading image data/one_28x28.pgm
Performing forward propagation ...
Testing cudnnGetConvolutionForwardAlgorithm ...
Fastest algorithm is Algo 1
Testing cudnnFindConvolutionForwardAlgorithm ...
^^^^ CUDNN_STATUS_SUCCESS for Algo 0: 0.033888 time requiring 0 memory
^^^^ CUDNN_STATUS_SUCCESS for Algo 1: 0.049472 time requiring 3464 memory
^^^^ CUDNN_STATUS_SUCCESS for Algo 2: 0.058400 time requiring 28800 memory
^^^^ CUDNN_STATUS_SUCCESS for Algo 5: 0.298688 time requiring 203008 memory
^^^^ CUDNN_STATUS_SUCCESS for Algo 4: 0.350624 time requiring 207360 memory
Resulting weights from Softmax:
0.0000001 1.0000000 0.0000001 0.0000000 0.0000563 0.0000001 0.0000012 0.0000017
0.0000010 0.0000001
Loading image data/three_28x28.pgm
Performing forward propagation ...
Resulting weights from Softmax:
0.0000000 0.0000000 0.0000000 1.0000000 0.0000000 0.0000714 0.0000000 0.0000000
0.0000000 0.0000000
Loading image data/five_28x28.pgm
Performing forward propagation ...
Resulting weights from Softmax:
0.0000000 0.0000008 0.0000000 0.0000002 0.0000000 1.0000000 0.0000154 0.0000000
0.0000012 0.0000006

Result of classification: 1 3 5

Test passed!
```

或者，输入以下命令来检查：

```
cat /usr/local/cuda/include/cudnn.h | grep CUDNN_MAJOR -A 2
```

如果显示如下图则说明成功



```
#define CUDNN_MAJOR 7
#define CUDNN_MINOR 6
#define CUDNN_PATCHLEVEL 5
--
#define CUDNN_VERSION (CUDNN_MAJOR * 1000 + CUDNN_MINOR * 100 + CUDNN_PATCHLEVEL
)

#include "driver_types.h"
```