1.打开terminal

2.进入黑名单目录

cd /etc/modprobe.d/

3.创建blacklist-nouveau.conf并添加内容

sudo touch blacklist-nouveau.conf

sudo echo "blacklist nouveau">>blacklist-nouveau.conf

sudo echo "options nouveau modeset=0">>blacklist-nouveau.conf

4.查看blacklist-nouveau.conf内容是否添加成功

sudo cat blacklist-nouveau.conf

5.更新blacklist

sudo update-initramfs -u

6.重启

sudo reboot

7.重启后查看驱动是否还存在，如果已经禁用了，则输出下面指令后没有任何输出

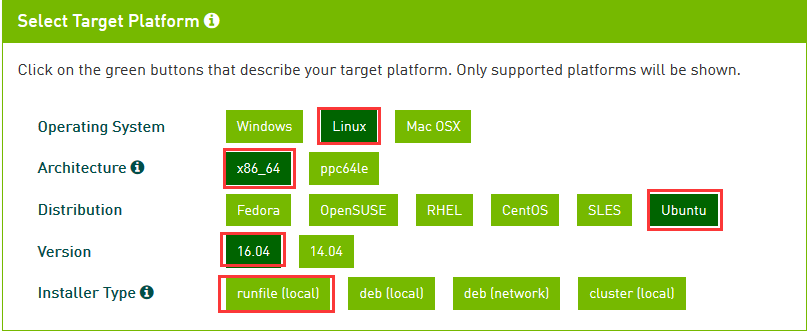
apt-get update

lsmod | grep nouveau

8.准备安装与显卡有关的内容，先装CUDA，再装显卡驱动，需要提前到NVIDIA官网下载，下载比较慢，请提前下载cudatoolkit和驱动：

Cudatoolkit：<https://developer.nvidia.com/cuda-downloads>

（Linux->x86\_64->ubuntu->16.04->runfile(local)）一定要下载runfile，别的文件格式装不好



显卡驱动：



9.安装CUDA，安装系统常用的包

sudo apt-get install vim vnc4server

10.安装cuda所需基本库

sudo apt-get install make gcc libssl-dev

11.安装cuda所需高级库，（注意：这个地方我有两个库没装上，提示找不到源，应该不影响）

sudo apt-get install build-essential freeglut3 freeglut3-dev libXi-dev libXmu-dev git

关闭图形界面

/etc/init.d/lightdm stop

12.CUDA安装，进入保存cuda的文件夹，使用cd进入即可：提示你要不要安装驱动，需要安装cuda自带显卡驱动就按Y，如有专用驱动，就按N（我们这里先不装驱动，后面单独装，所以按N），下面的根据提示按y（要不要安装tookit啥的）或者直接回车即可（安装路径）

Sudo bash cuda\_8.0.61\_375.26\_linux-run --no-opengl-libs

不报错 就好了

13.添加cuda的可执行程序的环境变量，environment里面原来格式是（ PATH=”path1:path2:path3” ），最后效果为（ PATH=”path1:path2:path3:/usr/local/cuda/bin” ）

sudo vim /etc/environment

添加 /usr/local/cuda/bin

14.添加cuda的library环境变量，添加 /usr/local/cuda/lib64 即可，没有别的格式

sudo vim /etc/ld.so.conf.d/cuda.conf

添加 /usr/local/cuda/lib64/

sudo ldconfig

让lib环境立刻生效

15.去/usr/local/lib/cuda-8.0/samples目录下

sudo make -j 16

完成后会生成bin目录

到最底层，跑算例./bandwidth

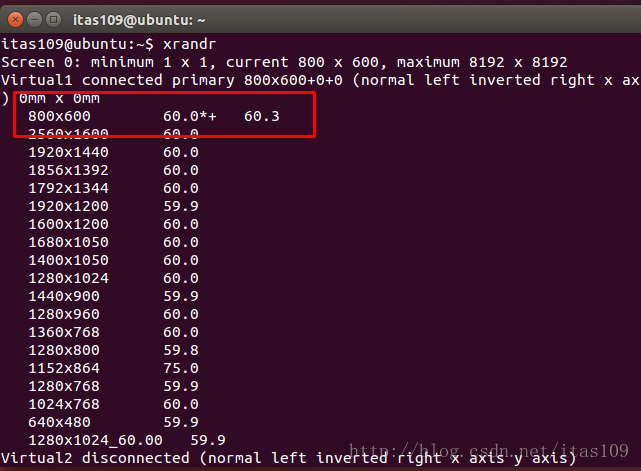
16.安装独立显卡驱动，进入保存驱动.run文件的文件夹下“”

装驱动执行：

sudo bash NVIDIA-Linux-x86\_64-375.66.run --no-opengl-files

安装完驱动后重启发现分辨率很不对，执行下面操作：

xrandr 带\*的为当前的分辨率，其他的为可选的分辨率设置



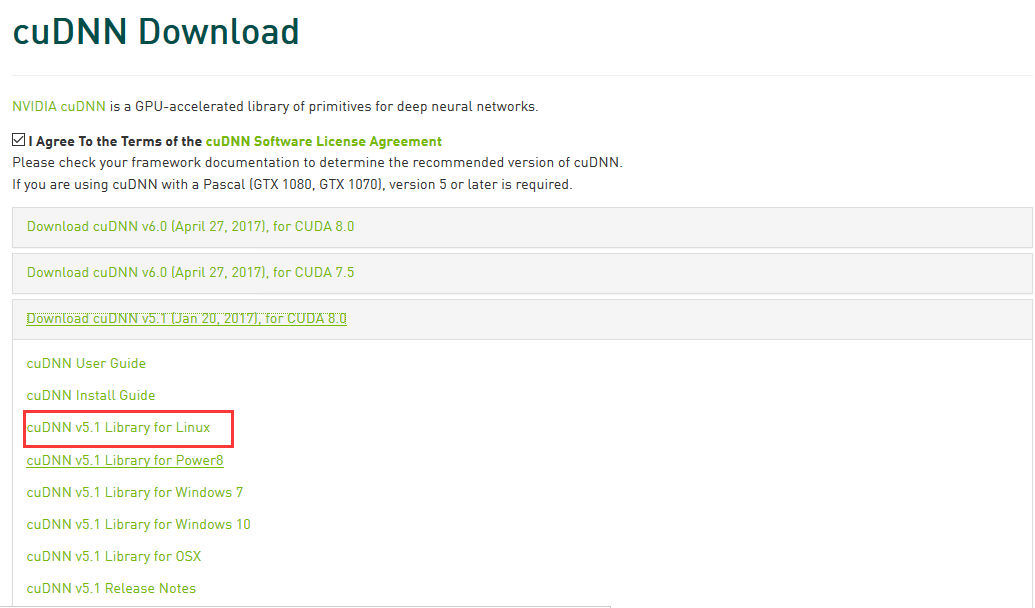
要是没装好，执行下面两句话可以把cuda和驱动卸载

sudo /usr/local/cuda-8.0/bin/uninstall\_cuda\_8.0.pl

sudo /usr/bin/nvidia-uninstall

17.添加cuDNN的库：

下载：https://developer.nvidia.com/rdp/cudnn-download



解压后，进入解压的文件夹，

cd cuda

sudo cp lib64/\* /usr/local/cuda/lib64/

sudo cp include/\* /usr/local/cuda/include/

完成cudnn的添加

更新cudnn（5更新到6）参考：[http://blog.csdn.net/suodislie/1article/details/70228138](http://blog.csdn.net/suodislie/article/details/70228138)

cudnn\_V7和v6兼容，安装方式同样是拷贝文件到指定文件夹下，同时设立软链接：

执行下列语句：

sudo rm -rf libcudnn.so.6 libcudnn.so

sudo ln -s libcudnn.so.7.0.5 libcudnn.so.7

sudo ln -s libcudnn.so.7 libcudnn.so

原来文件夹内的三个文件其实就libcudnn.so.7.0.5有用，其他的都是软链接生成出来的，使用ll指令就可以看出三个文件之间的关系，

18.安装tensorflow gpu支持版本

sudo apt-get install python-pip python-dev

sudo pip install --upgrade pip

sudo pip install --upgrade tensorflow-gpu

或者

sudo pip install –upgrade https://storage.googleapis.com/tensorflow/linux/gpu/tensorflow\_gpu-1.2.0-cp27-none-linux\_x86\_64.whl

卸载tensorflow的指令

**sudo pip uninstall tensorflow**